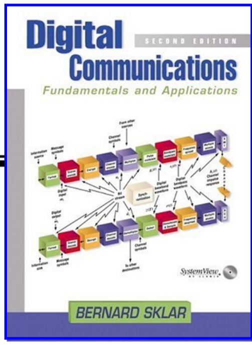


DIGITAL COMMUNICATIONS

Fundamentals and Applications

Second Edition

BERNARD SKLAR

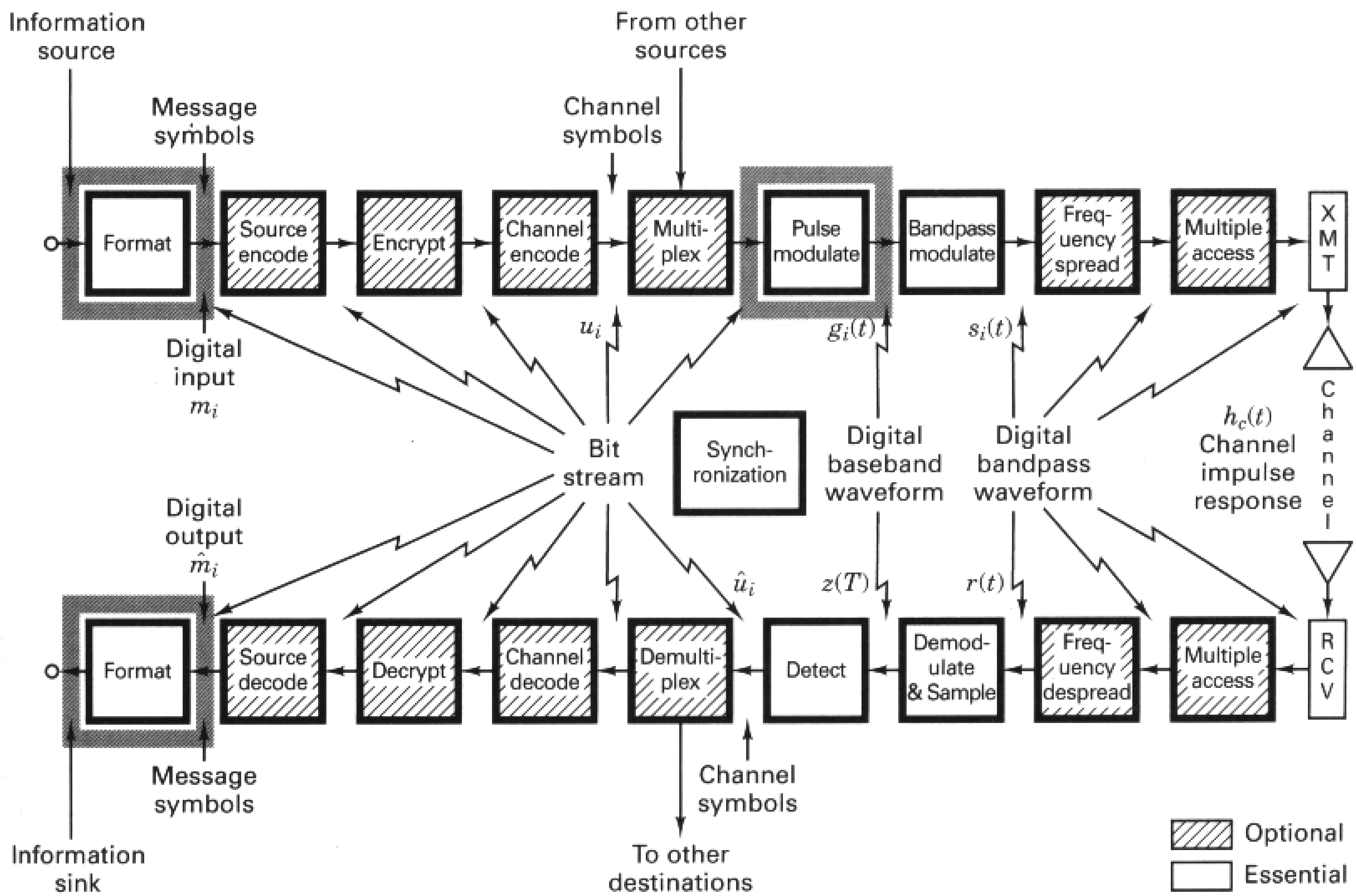
Communications Engineering Services, Tarzana, California
and
University of California, Los Angeles

PH PTR

ISBN 0-13-084788-7

Prentice Hall P T R
Upper Saddle River, New Jersey 07458
www.phptr.com

Formatting and Baseband Modulation



The goal of the first essential signal processing step, *formatting,* is to insure that the message (or source signal) is compatible with digital processing. *Transmit formatting* is a transformation from source information to digital symbols. (It is the reverse transformation in the receive chain.) When data compression in addition to formatting is employed, the process is termed *source coding.* Some authors consider formatting a special case of source coding. We treat formatting (and baseband modulation) in this chapter, and treat source coding as a special case of the *efficient description* of source information in Chapter 13.

In Figure 2.1, the highlighted block labeled "formatting" contains a list of topics that deal with transforming information to digital messages. The digital messages are considered to be in the logical format of binary ones and zeros until they are transformed by the next essential step, called pulse modulation, into *baseband* (pulse) waveforms. Such waveforms can then be transmitted over a cable. In Figure 2.1, the highlighted block labeled "baseband signaling" contains a list of pulse modulating waveforms that are described in this chapter. The term baseband refers to a signal whose spectrum extends from (or near) dc up to some finite value, usually less than a few megahertz. In Chapter 3, the subject of baseband signaling is continued with emphasis on demodulation and detection.

2.1 BASEBAND SYSTEMS

In Figure 1.2 we presented a block diagram of a typical digital communication system. A version of this functional diagram, focusing primarily on the formatting and transmission of *baseband* signals, is shown in Figure 2.2. Data already in a digital

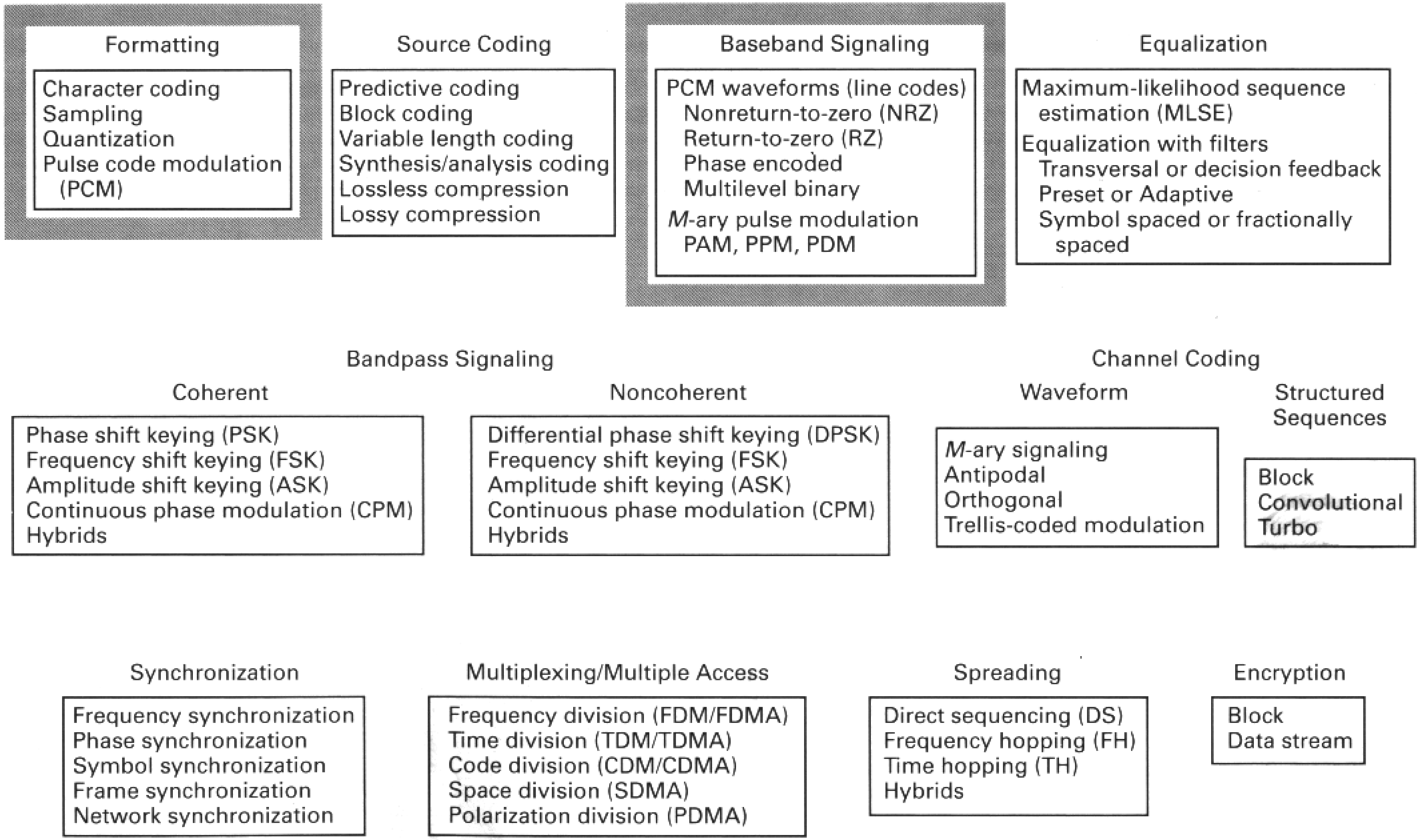


Figure 2.1 Basic digital communication transformations

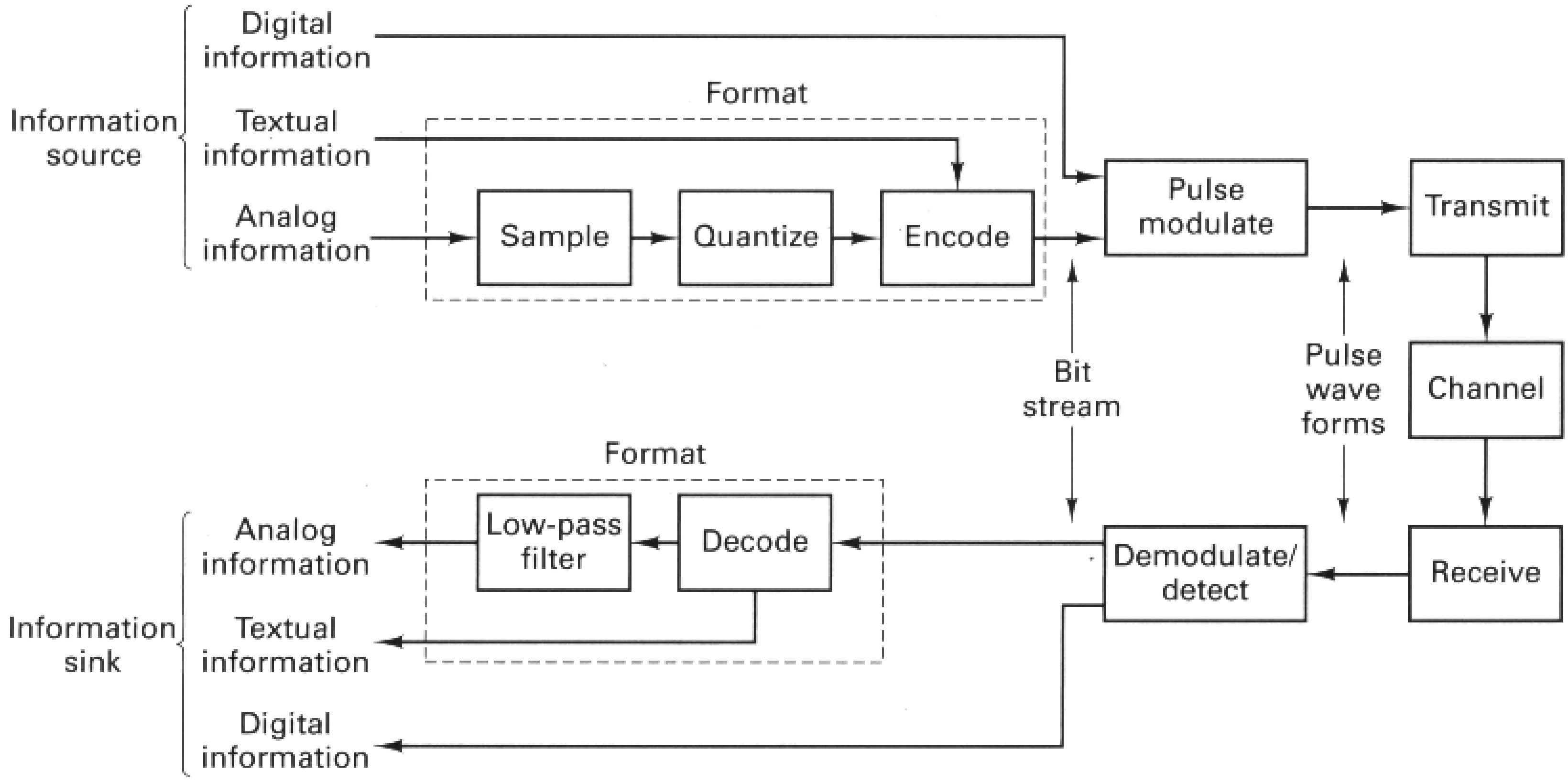


Figure 2.2 Formatting and transmission of baseband signals.

format would bypass the formatting function. Textual information is transformed into binary digits by use of a coder. Analog information is formatted using three separate processes: sampling, quantization, and coding. In all cases, the formatting step results in a sequence of binary digits.

These digits are to be transmitted through a *baseband channel,* such as a pair of wires or a coaxial cable. However, no channel can be used for the transmission of binary digits without first transforming the digits to *waveforms* that are compatible with the channel. For baseband channels, compatible waveforms are pulses.

In Figure 2.2, the conversion from a bit stream to a sequence of pulse waveforms takes place in the block labeled pulse modulate. The output of the modulator is typically a sequence of pulses with characteristics that correspond to the digits being sent. After transmission through the channel, the pulse waveforms are recovered (demodulated) and detected to produce an estimate of the transmitted digits; the final step, (reverse) formatting, recovers an estimate of the source information.

2.2 FORMATTING TEXTUAL DATA (CHARACTER CODING)

The original form of most communicated data (except for computer-to-computer transmissions) is either textual or analog. If the data consist of alphanumeric text, they will be character encoded with one of several standard formats; examples include the American Standard Code for Information Interchange (ASCII), the Extended Binary Coded Decimal Interchange Code (EBCDIC), Baudot, and Hollerith. The textual material is thereby transformed into a digital format. The ASCII format is shown in Figure 2.3; the EBCDIC format is shown in Figure 2.4.

Bits											
				5	0	1	0	1	0	1	0	1
				6	0	0	1	1	0	0	1	1
1	2	3	4	7	0	0	0	0	1	1	1	1
0	0	0	0		NUL	DLE	SP	0	@	P	'	p
1	0	0	0		SOH	DC1	!	1	A	Q	a	q
0	1	0	0		STX	DC2	"	2	B	R	b	r
1	1	0	0		ETX	DC3	#	3	C	S	c	s
0	0	1	0		EOT	DC4	$	4	D	T	d	t
1	0	1	0		ENQ	NAK	%	5	E	U	e	u
0	1	1	0		ACK	SYN	&	6	F	V	f	v
1	1	1	0		BEL	ETB	'	7	G	W	g	w
0	0	0	1		BS	CAN	(	8	H	X	h	x
1	0	0	1		HT	EM	)	9	I	Y	i	y
0	1	0	1		LF	SUB	*	:	J	Z	j	z
1	1	0	1		VT	ESC	+	;	K	[	k	{
0	0	1	1		FF	FS	,	<	L	\	l	|
1	0	1	1		CR	GS	–	=	M	]	m	}
0	1	1	1		SO	RS	.	>	N	^	n	~
1	1	1	1		SI	US	/	?	O	–	o	DEL

NUL	Null, or all zeros	DC1	Device control 1
SOH	Start of heading	DC2	Device control 2
STX	Start of text	DC3	Device control 3
ETX	End of text	DC4	Device control 4
EOT	End of transmission	NAK	Negative acknowledge
ENQ	Enquiry	SYN	Synchronous idle
ACK	Acknowledge	ETB	End of transmission
BEL	Bell, or alarm	CAN	Cancel
BS	Backspace	EM	End of medium
HT	Horizontal tabulation	SUB	Substitute
LF	Line feed	ESC	Escape
VT	Vertical tabulation	FS	File separator
FF	Form feed	GS	Group separator
CR	Carriage return	RS	Record separator
SO	Shift out	US	Unit separator
SI	Shift in	SP	Space
DLE	Data link escape	DEL	Delete

Figure 2.3 Seven-bit American standard code for information interchange (ASCII).

Bits	5	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	6	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	7	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	8	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
1 2 3 4																	
0 0 0 0		NUL	SOH	STX	ETX	PF	HT	LC	DEL			SMM	VT	FF	CR	SO	SI
0 0 0 1		DLE	DC1	DC2	DC3	RES	NL	BS	IL	CAN	EM	CC		IFS	IGS	IRS	IUS
0 0 1 0		DS	SOS	FS		BYP	LF	EOB	PRE			SM			ENQ	ACK	BEL
0 0 1 1				SYN		PN	RS	US	EOT					DC4	NAK		SUB
0 1 0 0		SP										¢		<	(	+	!
0 1 0 1		&										!	$	*	)	;	¬
0 1 1 0		–	/										,	%	_	>	?
0 1 1 1												:	#	@	'	=	"
1 0 0 0			a	b	c	d	e	f	g	h	i						
1 0 0 1			j	k	l	m	n	o	p	q	r						
1 0 1 0				s	t	u	v	w	x	y	z						
1 0 1 1																	
1 1 0 0			A	B	C	D	E	F	G	H	I						
1 1 0 1			J	K	L	M	N	O	P	Q	R						
1 1 1 0				S	T	U	V	W	X	Y	Z						
1 1 1 1		0	1	2	3	4	5	6	7	8	9						

PF	Punch off
HT	Horizontal tab
LC	Lower case
DEL	Delete
SP	Space
UC	Upper case
RES	Restore
NL	New line
BS	Backspace
IL	Idle
PN	Punch on
EOT	End of transmission
BYP	Bypass
LF	Line feed
EOB	End of block
PRE	Prefix (ESC)
RS	Reader stop
SM	Start message
DS	Digit select
SOS	Start of significance
IFS	Interchange file separator
IGS	Interchange group separator
IRS	Interchange record separator
IUS	Interchange unit separator
Others	Same as ASCII

Figure 2.4 EBCDIC character code set.

The bit numbers signify the order of serial transmission, where bit number 1 is the first signaling element. Character coding, then, is the step that transforms text into binary digits (bits). Sometimes existing character codes are modified to meet specialized needs. For example, the 7-bit ASCII code (Figure 2.3) can be modified to include an added bit for error detection purposes. (See Chapter 6.) On the other hand, sometimes the code is truncated to a 6-bit ASCII version, which provides capability for only 64 characters instead of the 128 characters allowed by 7-bit ASCII.

2.3 MESSAGES, CHARACTERS, AND SYMBOLS

Textual messages comprise a sequence of alphanumeric characters. When digitally transmitted, the characters are first encoded into a sequence of bits, called a *bit stream* or *baseband signal.* Groups of k bits can then be combined to form new digits, or *symbols,* from a finite symbol set or alphabet of $M = 2^k$ such symbols. A system using a symbol set size of M is referred to as an *M-ary system.* The value of k or M represents an important initial choice in the design of any digital communication system. For $k = 1$, the system is termed *binary,* the size of the symbol set is $M = 2$, and the modulator uses one of the two different waveforms to represent the binary "one" and the other to represent the binary "zero." For this special case, the symbol and the bit are the same. For $k = 2$, the system is termed *quaternary* or *4-ary* ($M = 4$). At each symbol time, the modulator uses one of the four different waveforms that represents the symbol. The partitioning of the sequence of message bits is determined by the specification of the symbol set size, M. The following example should help clarify the relationship between the following terms: "message," "character," "symbol," "bit," and "digital waveform."

2.3.1 Example of Messages, Characters, and Symbols

Figure 2.5 shows examples of bit stream partitioning, based on the system specification for the values of k and M. The textual message in the figure is the word "THINK." Using 6-bit ASCII character coding (bit numbers 1 to 6 from Figure 2.3) yields a bit stream comprising 30 bits. In Figure 2.5a, the symbol set size, M, has been chosen to be 8 (each symbol represents an 8-ary digit). The bits are therefore partitioned into groups of three ($k = \log_2 8$); the resulting 10 numbers represent the 10 octal symbols to be transmitted. The transmitter must have a repertoire of eight waveforms $s_i(t)$, where $i = 1, \ldots, 8$, to represent the possible symbols, any one of which may be transmitted during a symbol time. The final row of Figure 2.5a lists the 10 waveforms that an 8-ary modulating system transmits to represent the textual message "THINK."

In Figure 2.5b, the symbol set size, M, has been chosen to be 32 (each symbol represents a 32-ary digit). The bits are therefore taken five at a time, and the resulting group of six numbers represent the six 32-ary symbols to be transmitted. Notice that there is no need for the symbol boundaries and the character boundaries to coincide. The first symbol represents $\frac{5}{6}$ of the first character, "T." The second symbol

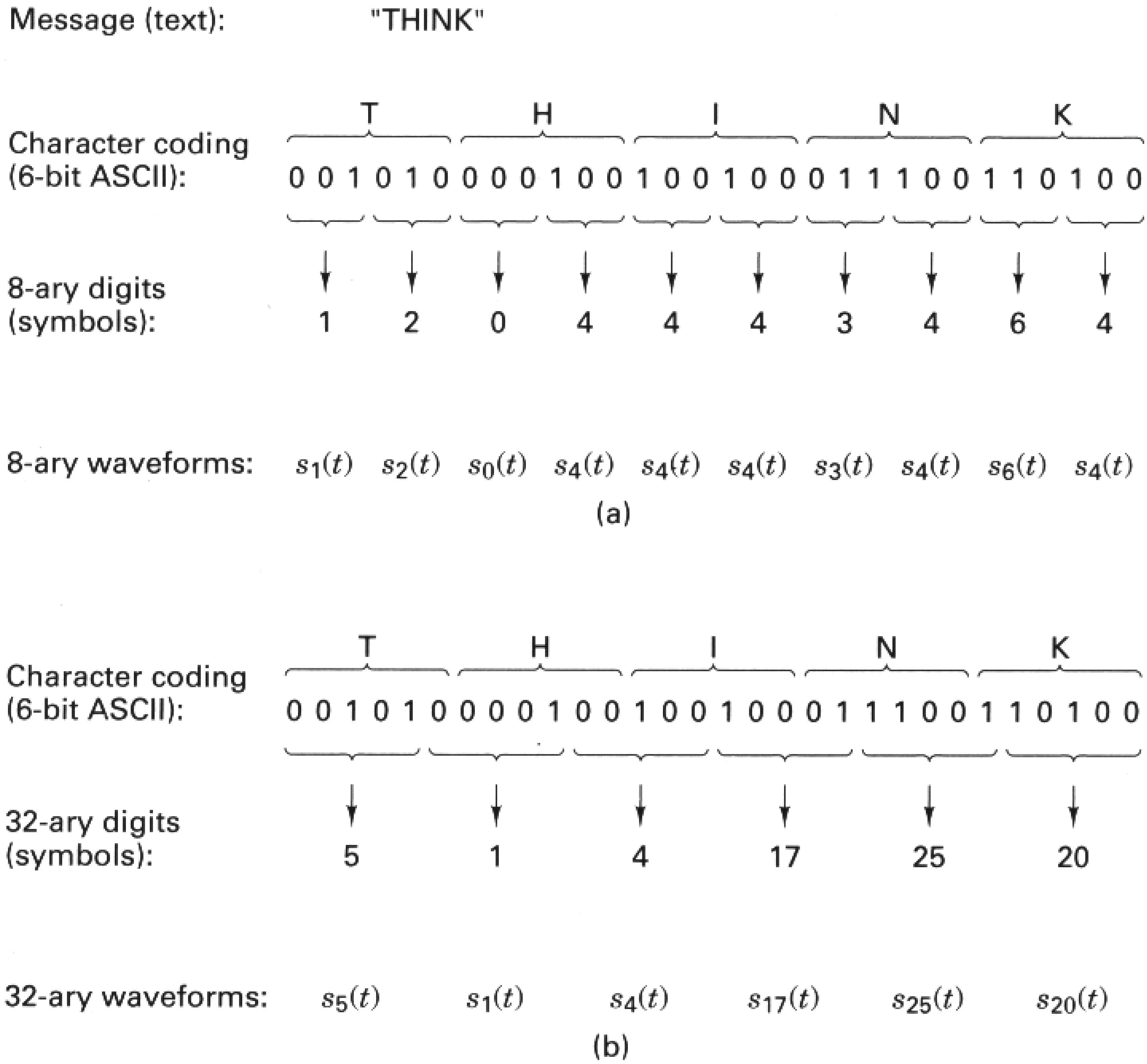


Figure 2.5 Messages, characters, and symbols. (a) 8-ary example. (b) 32-ary example.

represents the remaining $\frac{1}{6}$ of the character "T" and $\frac{4}{6}$ of the next character, "H," and so on. It is not necessary that the characters be partitioned more aesthetically. The system sees the characters as a string of digits to be transmitted; only the end user (or the user's teleprinter machine) ascribes textual meaning to the final delivered sequence of bits. In this 32-ary case, a transmitter needs a repertoire of 32 waveforms $s_i(t)$, where $i = 1, \ldots, 32$, one for each possible symbol that may be transmitted. The final row of the figure lists the six waveforms that a 32-ary modulating system transmits to represent the textual message "THINK."

2.4 FORMATTING ANALOG INFORMATION

If the information is analog, it cannot be character encoded as in the case of textual data; the information must first be transformed into a digital format. The process of transforming an analog waveform into a form that is compatible with a digital com-

munication system starts with sampling the waveform to produce a discrete pulse-amplitude-modulated waveform, as described below.

2.4.1 The Sampling Theorem

The link between an analog waveform and its sampled version is provided by what is known as the *sampling process.* This process can be implemented in several ways, the most popular being the *sample-and-hold* operation. In this operation, a switch and storage mechanism (such as a transistor and a capacitor, or a shutter and a filmstrip) form a sequence of samples of the continuous input waveform. The output of the sampling process is called *pulse amplitude modulation* (PAM) because the successive output intervals can be described as a sequence of pulses with amplitudes derived from the input waveform samples. The analog waveform can be approximately retrieved from a PAM waveform by simple low-pass filtering. An important question: how closely can a filtered PAM waveform approximate the original input waveform? This question can be answered by reviewing the *sampling theorem,* which states the following [1]: A bandlimited signal having no spectral components above f_m hertz can be determined uniquely by values sampled at uniform intervals of

$$T_s \leq \frac{1}{2f_m} \text{ sec} \tag{2.1}$$

This particular statement is also known as the *uniform sampling theorem.* Stated another way, the upper limit on T_s can be expressed in terms of the sampling rate, denoted $f_s = 1/T_s$. The restriction, stated in terms of the sampling rate, is known as the *Nyquist criterion.* The statement is

$$f_s \geq 2f_m \tag{2.2}$$

The sampling rate $f_s = 2f_m$ is also called the *Nyquist rate.* The Nyquist criterion is a theoretically sufficient condition to allow an analog signal to be *reconstructed completely* from a set of uniformly spaced discrete-time samples. In the sections that follow, the validity of the sampling theorem is demonstrated using different sampling approaches.

2.4.1.1 Impulse Sampling

Here we demonstrate the validity of the sampling theorem using the frequency convolution property of the Fourier transform. Let us first examine the case of *ideal sampling* with a sequence of unit impulse functions. Assume an analog waveform, $x(t)$, as shown in Figure 2.6a, with a Fourier transform, $X(f)$, which is zero outside the interval $(-f_m < f < f_m)$, as shown in Figure 2.6b. The sampling of $x(t)$ can be viewed as the product of $x(t)$ with a periodic train of unit impulse functions $x_\delta(t)$, shown in Figure 2.6c and defined as

$$x_\delta(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \tag{2.3}$$

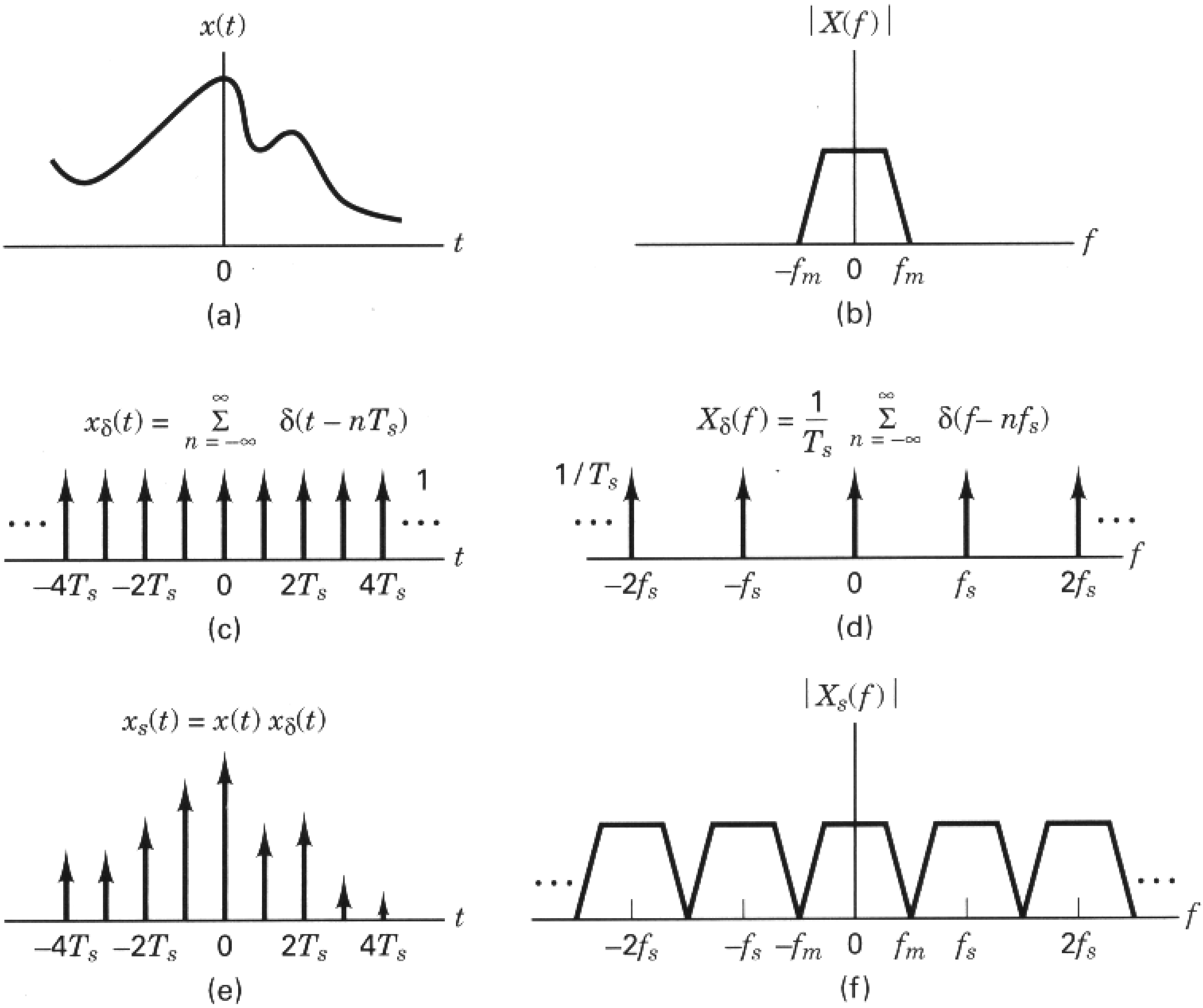


Figure 2.6 Sampling theorem using the frequency convolution property of the Fourier transform.

where T_s is the sampling period and $\delta(t)$ is the unit impulse or Dirac delta function defined in Section 1.2.5. Let us choose $T_s = 1/2f_m$, so that the Nyquist criterion is just satisfied.

The *sifting property* of the impulse function (see Section A.4.1) states that

$$x(t)\delta(t - t_0) = x(t_0)\delta(t - t_0) \tag{2.4}$$

Using this property, we can see that $x_s(t)$, the sampled version of $x(t)$ shown in Figure 2.6e, is given by

$$\begin{aligned} x_s(t) = x(t)x_\delta(t) &= \sum_{n=-\infty}^{\infty} x(t)\delta(t - nT_s) \\ &= \sum_{n=-\infty}^{\infty} x(nT_s)\delta(t - nT_s) \end{aligned} \tag{2.5}$$

Using the *frequency convolution property* of the Fourier transform (see Section A.5.3), we can transform the time-domain product $x(t)x_\delta(t)$ of Equation (2.5) to the frequency-domain convolution $X(f) * X_\delta(f)$, where

$$X_\delta(f) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} \delta(f - nf_s) \tag{2.6}$$

is the Fourier transform of the impulse train $x_\delta(t)$ and where $f_s = 1/T_s$ is the sampling frequency. Notice that the Fourier transform of an impulse train is another impulse train; the values of the periods of the two trains are reciprocally related to one another. Figures 2.6c and d illustrate the impulse train $x_\delta(t)$ and its Fourier transform $X_\delta(f)$, respectively.

Convolution with an impulse function simply shifts the original function as follows:

$$X(f) * \delta(f - nf_s) = X(f - nf_s) \tag{2.7}$$

We can now solve for the transform $X_s(f)$ of the sampled waveform:

$$\begin{aligned} X_s(f) = X(f) * X_\delta(f) &= X(f) * \left[\frac{1}{T_s} \sum_{n=-\infty}^{\infty} \delta(f - nf_s)\right] \\ &= \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X(f - nf_s) \end{aligned} \tag{2.8}$$

We therefore conclude that within the original bandwidth, the spectrum $X_s(f)$ of the sampled signal $x_s(t)$ is, to within a constant factor $(1/T_s)$, exactly the same as that of $x(t)$. In addition, the spectrum repeats itself periodically in frequency every f_s hertz. The sifting property of an impulse function makes the convolving of an impulse train with another function easy to visualize. The impulses act as sampling functions. Hence, convolution can be performed graphically by sweeping the impulse train $X_\delta(f)$ in Figure 2.6d past the transform $|X(f)|$ in Figure 2.6b. This sampling of $|X(f)|$ at each step in the sweep replicates $|X(f)|$ at each of the frequency positions of the impulse train, resulting in $|X_s(f)|$, shown in Figure 2.6f.

When the sampling rate is chosen, as it has been here, such that $f_s = 2f_m$, each spectral replicate is separated from each of its neighbors by a frequency band exactly equal to f_s hertz, and the analog waveform can theoretically be completely recovered from the samples, by the use of filtering. However, a filter with infinitely steep sides would be required. It should be clear that if $f_s > 2f_m$, the replications will move farther apart in frequency, as shown in Figure 2.7a, making it easier to perform the filtering operation. A typical low-pass filter characteristic that might be used to separate the baseband spectrum from those at higher frequencies is shown in the figure. When the sampling rate is reduced, such that $f_s < 2f_m$, the replications will overlap, as shown in Figure 2.7b, and some information will be lost. The phenomenon, the result of undersampling (sampling at too low a rate), is called *aliasing*. The Nyquist rate, $f_s = 2f_m$, is the sampling rate below which aliasing occurs; to avoid aliasing, the Nyquist criterion, $f_s \geq 2f_m$, must be satisfied.

As a matter of practical consideration, neither waveforms of engineering interest nor realizable bandlimiting filters are strictly bandlimited. Perfectly bandlimited signals do not occur in nature (see Section 1.7.2); thus, realizable signals, even though we may think of them as bandlimited, always contain some aliasing. These signals and filters can, however, be considered to be "essentially" bandlimited. By

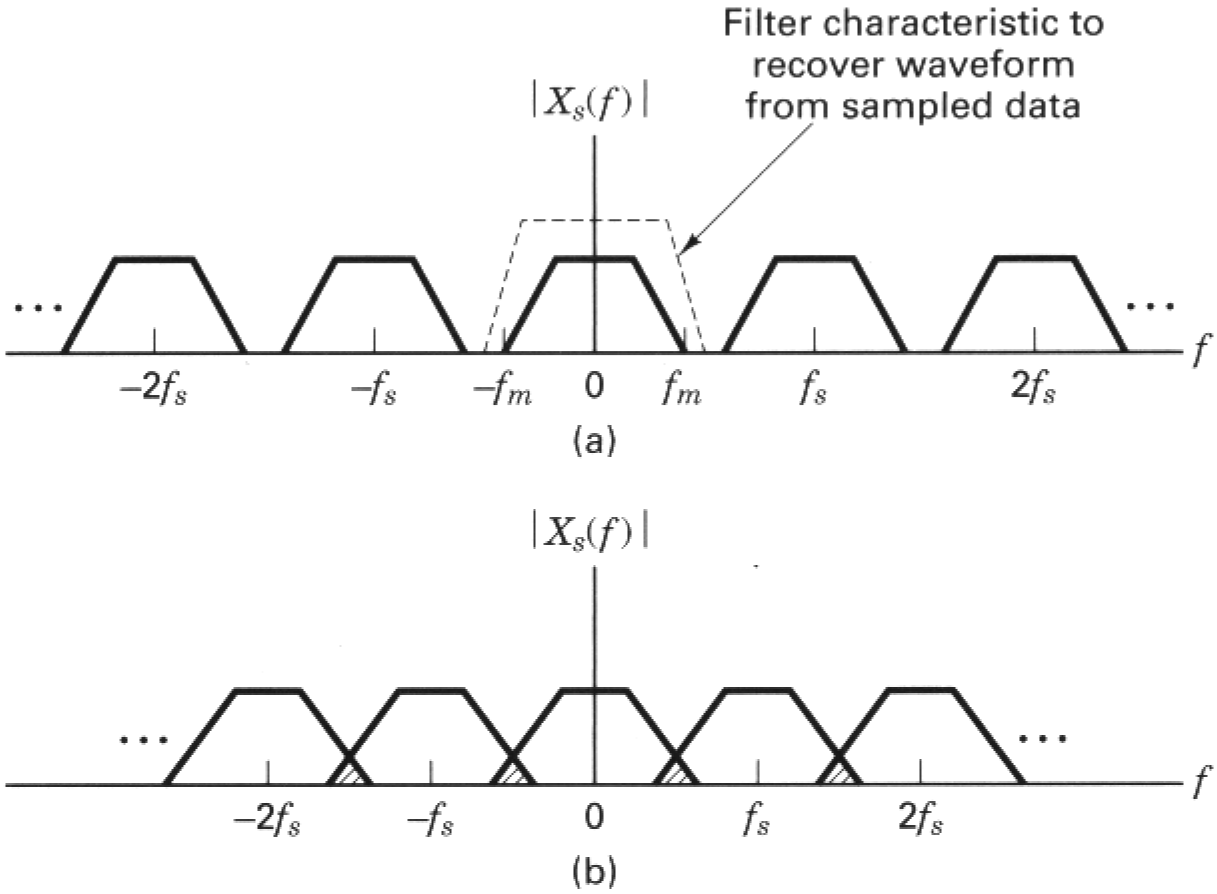


Figure 2.7 Spectra for various sampling rates. (a) Sampled spectrum ($f_s > 2f_m$). (b) Sampled spectrum ($f_s < 2f_m$).

this we mean that a bandwidth can be determined beyond which the spectral components are attenuated to a level that is considered negligible.

2.4.1.2 Natural Sampling

Here we demonstrate the validity of the sampling theorem using the frequency shifting property of the Fourier transform. Although instantaneous sampling is a convenient model, a more practical way of accomplishing the sampling of a bandlimited analog signal $x(t)$ is to multiply $x(t)$, shown in Figure 2.8a, by the pulse train or switching waveform $x_p(t)$, shown in Figure 2.8c. Each pulse in $x_p(t)$ has width T and amplitude $1/T$. Multiplication by $x_p(t)$ can be viewed as the opening and closing of a switch. As before, the sampling frequency is designated f_s, and its reciprocal, the time period between samples, is designated T_s. The resulting sampled-data sequence, $x_s(t)$, is illustrated in Figure 2.8e and is expressed as

$$x_s(t) = x(t)x_p(t) \tag{2.9}$$

The sampling here is termed *natural sampling,* since the top of each pulse in the $x_s(t)$ sequence retains the shape of its corresponding analog segment during the pulse interval. Using Equation (A.13), we can express the periodic pulse train as a Fourier series in the form

$$x_p(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\,2\pi n f_s t} \tag{2.10}$$

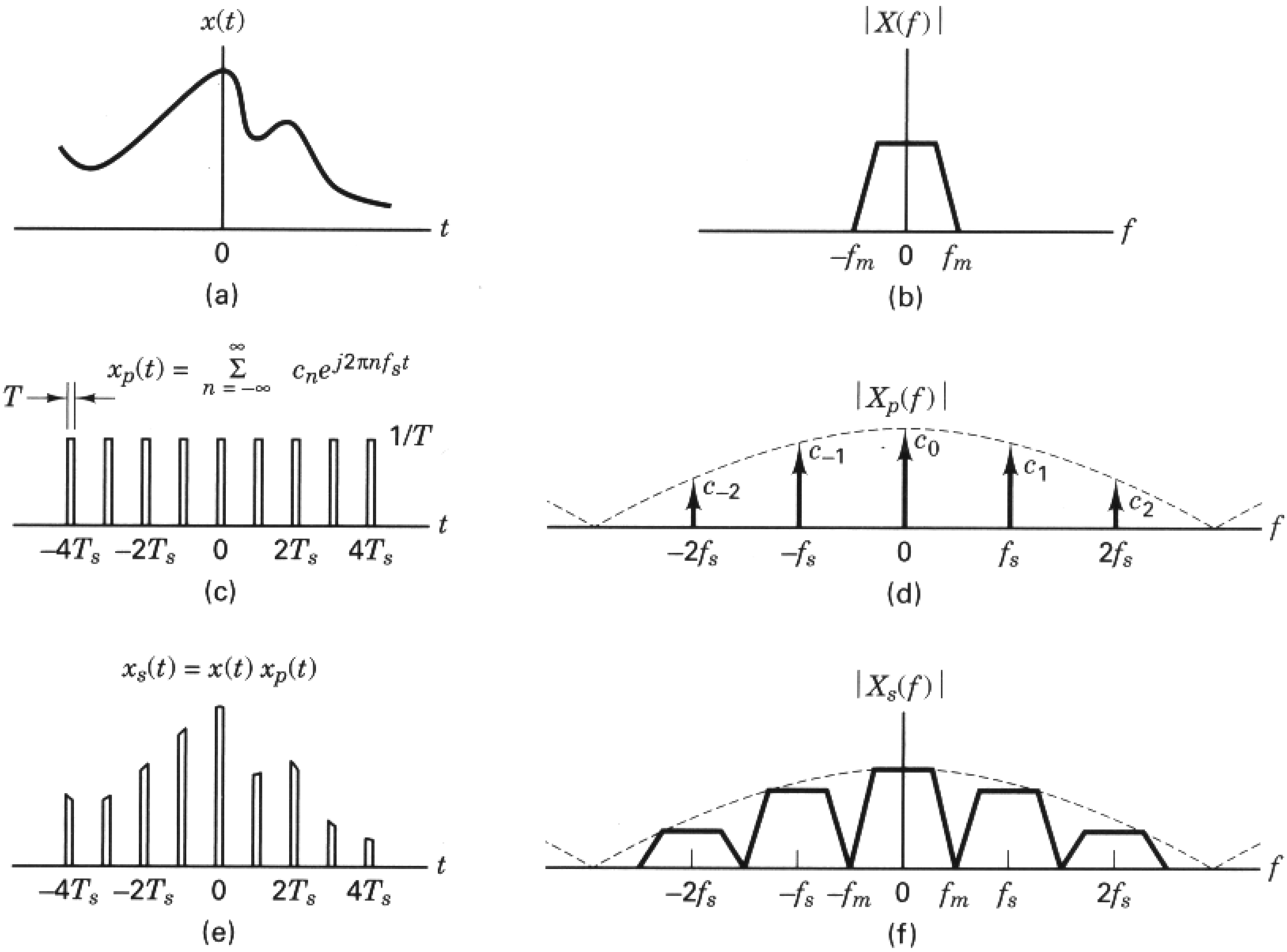


Figure 2.8 Sampling theorem using the frequency shifting property of the Fourier transform.

where the sampling rate, $f_s = 1/T_s$, is chosen equal to $2f_m$, so that the Nyquist criterion is just satisfied. From Equation (A.24), $c_n = (1/T_s)\ \mathrm{sinc}\ (nT/T_s)$, where T is the pulse width, $1/T$ is the pulse amplitude, and

$$\mathrm{sinc}\ y = \frac{\sin \pi y}{\pi y}$$

The envelope of the magnitude spectrum of the pulse train, seen as a dashed line in Figure 2.8d, has the characteristic sinc shape. Combining Equations (2.9) and (2.10) yields

$$x_s(t) = x(t) \sum_{n=-\infty}^{\infty} c_n e^{j\,2\pi n f_s t} \tag{2.11}$$

The transform $X_s(f)$ of the sampled waveform is found as follows:

$$X_s(f) = \mathcal{F}\left\{x(t) \sum_{n=-\infty}^{\infty} c_n e^{j\,2\pi n f_s t}\right\} \tag{2.12}$$

For linear systems, we can interchange the operations of summation and Fourier transformation. Therefore, we can write

$$X_s(f) = \sum_{n=-\infty}^{\infty} c_n \mathscr{F}\{x(t)e^{j2\pi nf_s t}\} \tag{2.13}$$

Using the *frequency translation* property of the Fourier transform (see Section A.3.2), we solve for $X_s(f)$ as follows:

$$X_s(f) = \sum_{n=-\infty}^{\infty} c_n X(f - nf_s) \tag{2.14}$$

Similar to the unit impulse sampling case, Equation (2.14) and Figure 2.8f illustrate that $X_s(f)$ is a replication of $X(f)$, periodically repeated in frequency every f_s hertz. In this natural-sampled case, however, we see that $X_s(f)$ is weighted by the Fourier series coefficients of the pulse train, compared with a constant value in the impulse-sampled case. It is satisfying to note that *in the limit,* as the pulse width, T, approaches zero, c_n approaches $1/T_s$ for all n (see the example that follows), and Equation (2.14) converges to Equation (2.8).

Example 2.1 Comparison of Impulse Sampling and Natural Sampling

Consider a given waveform $x(t)$ with Fourier transform $X(f)$. Let $X_{s1}(f)$ be the spectrum of $x_{s1}(t)$, which is the result of sampling $x(t)$ with a unit impulse train $x_\delta(t)$. Let $X_{s2}(f)$ be the spectrum of $x_{s2}(t)$, the result of sampling $x(t)$ with a pulse train $x_p(t)$ with pulse width T, amplitude $1/T$, and period T_s. Show that in the limit, as T approaches zero, $X_{s1}(f) = X_{s2}(f)$.

Solution

From Equation (2.8),

$$X_{s1}(f) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X(f - nf_s)$$

and from Equation (2.14),

$$X_{s2}(f) = \sum_{n=-\infty}^{\infty} c_n X(f - nf_s)$$

As the pulse with $T \rightarrow 0$, and the pulse amplitude approaches infinity (the area of the pulse remains unity), $x_p(t) \rightarrow x_\delta(t)$. Using Equation (A.14), we can solve for c_n in the limit as follows:

$$\begin{aligned} c_n &= \lim_{T\to 0} \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} x_p(t)e^{-j2\pi nf_s t}\,dt \\ &= \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} x_\delta(t)e^{-j2\pi nf_s t}\,dt \end{aligned}$$

Since, within the range of integration, $-T_s/2$ to $T_s/2$, the only contribution of $x_\delta(t)$ is that due to the impulse at the origin, we can write

$$c_n = \frac{1}{T_s}\int_{-T_s/2}^{T_s/2} \delta(t)e^{-j2\pi n f_s t}\,dt = \frac{1}{T_s}$$

Therefore, in the limit, $X_{s1}(f) = X_{s2}(f)$ for all n.

2.4.1.3 Sample-and-Hold Operation

The simplest and thus most popular sampling method, *sample and hold,* can be described by the convolution of the sampled pulse train, $[x(t)x_\delta(t)]$, shown in Figure 2.6e, with a unity amplitude rectangular pulse $p(t)$ of pulse width T_s. This time, convolution results in the *flattop* sampled sequence

$$\begin{aligned} x_s(t) &= p(t) * [x(t)x_\delta(t)] \\ &= p(t) * \left[x(t) \sum_{n=-\infty}^{\infty} \delta(t - nT_s)\right] \end{aligned} \tag{2.15}$$

The Fourier transform, $X_s(f)$, of the time convolution in Equation (2.15) is the frequency-domain product of the transform $P(f)$ of the rectangular pulse and the periodic spectrum, shown in Figure 2.6f, of the impulse-sampled data:

$$\begin{aligned} X_s(f) &= P(f)\mathcal{F}\left\{x(t) \sum_{n=-\infty}^{\infty} \delta(t - nT_s)\right\} \\ &= P(f)\left\{X(f) * \left[\frac{1}{T_s} \sum_{n=-\infty}^{\infty} \delta(f - nf_s)\right]\right\} \\ &= P(f)\frac{1}{T_s} \sum_{n=-\infty}^{\infty} X(f - nf_s) \end{aligned} \tag{2.16}$$

Here, $P(f)$ is of the form T_s sinc fT_s. The effect of this product operation results in a spectrum similar in appearance to the natural-sampled example presented in Figure 2.8f. The most obvious effect of the hold operation is the significant attenuation of the higher-frequency spectral replicates (compare Figure 2.8f to Figure 2.6f), which is a desired effect. Additional analog postfiltering is usually required to finish the filtering process by further attenuating the residual spectral components located at the multiples of the sample rate. A secondary effect of the hold operation is the nonuniform spectral gain $P(f)$ applied to the desired baseband spectrum shown in Equation (2.16). The postfiltering operation can compensate for this attenuation by incorporating the inverse of $P(f)$ over the signal passband.

2.4.2 Aliasing

Figure 2.9 is a detailed view of the positive half of the baseband spectrum and one of the replicates from Figure 2.7b. It illustrates aliasing in the frequency domain. The overlapped region, shown in Figure 2.9b, contains that part of the spectrum which is aliased due to *undersampling.* The aliased spectral components represent ambiguous data that appear in the frequency band between $(f_s - f_m)$ and f_m. Figure 2.10 illustrates that a higher sampling rate f'_s, can eliminate the aliasing by separat-

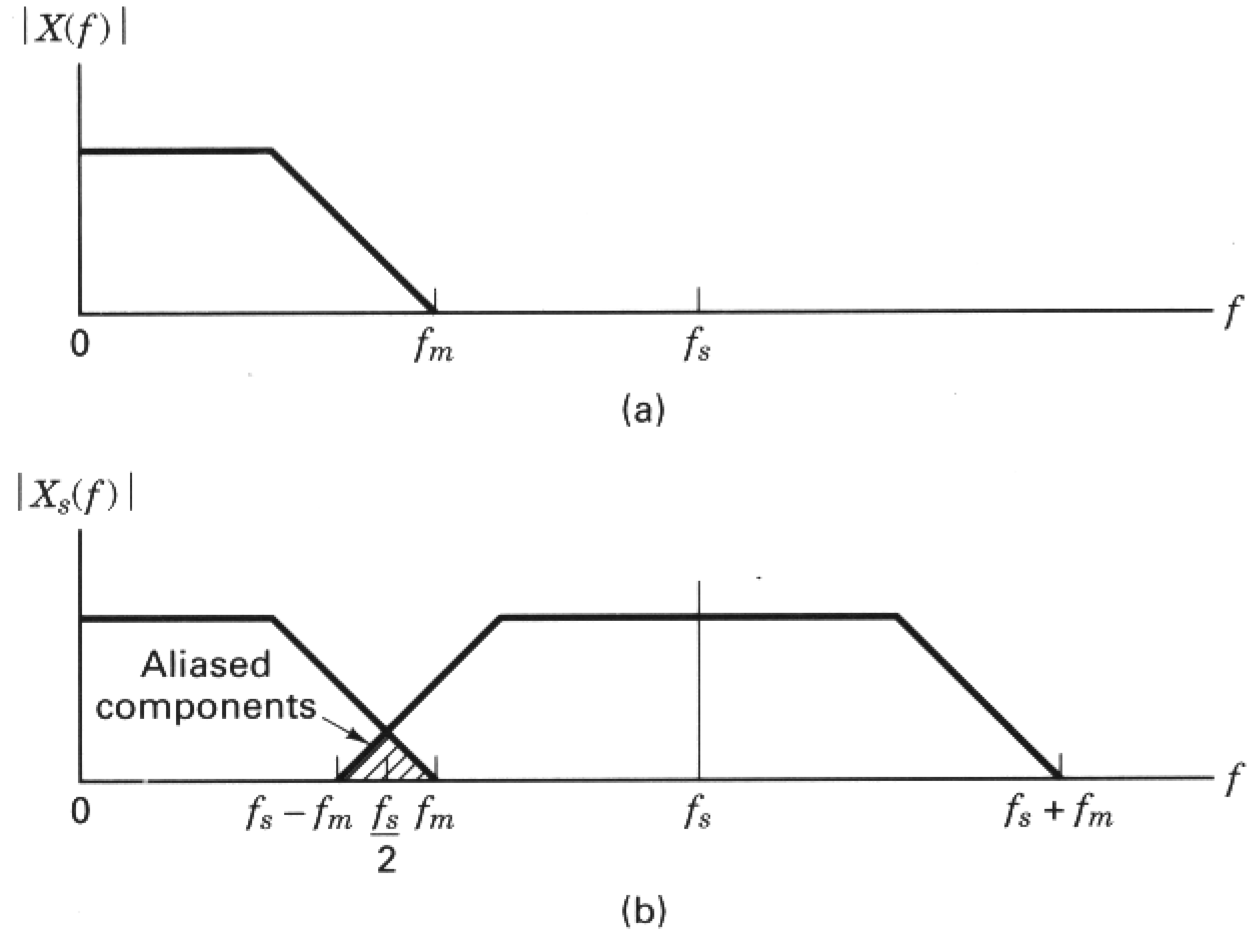


Figure 2.9 Aliasing in the frequency domain. (a) Continuous signal spectrum. (b) Sampled signal spectrum.

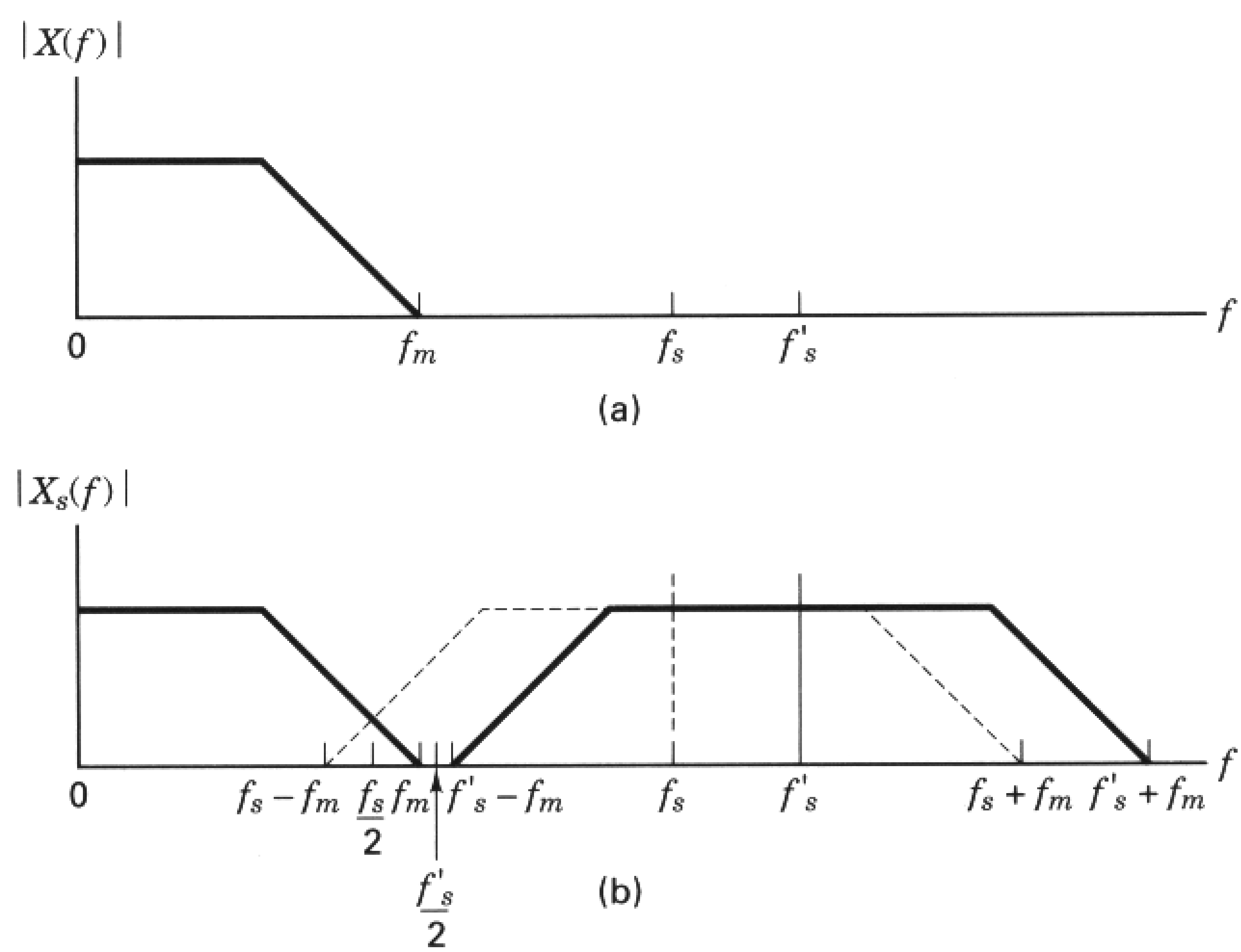


Figure 2.10 Higher sampling rate eliminates aliasing. (a) Continuous signal spectrum. (b) Sampled signal spectrum.

ing the spectral replicates; the resulting spectrum in Figure 2.10b corresponds to the case in Figure 2.7a. Figures 2.11 and 2.12 illustrate two ways of eliminating aliasing using *antialiasing filters.* In Figure 2.11 the analog signal is *prefiltered* so that the new maximum frequency, f'_m, is reduced to $f_s/2$ or less. Thus there are no aliased components seen in Figure 2.11b, since $f_s > 2f'_m$. Eliminating the aliasing terms prior to sampling is good engineering practice. When the signal structure is well known, the aliased terms can be eliminated after sampling, with a low-pass filter operating on the sampled data [2]. In Figure 2.12 the aliased components are removed by *postfiltering* after sampling; the filter cutoff frequency, f''_m, removes the aliased components; f''_m needs to be less than $(f_s - f_m)$. Notice that the filtering techniques for eliminating the aliased portion of the spectrum in Figures 2.11 and 2.12 *will result in a loss* of some of the signal information. For this reason, the sample rate, cutoff bandwidth, and filter type selected for a particular signal bandwidth are all interrelated.

Realizable filters require a nonzero bandwidth for the transition between the passband and the required out-of-band attenuation. This is called the *transition bandwidth.* To minimize the system sample rate, we desire that the antialiasing filter have a small transition bandwidth. Filter complexity and cost rise sharply with narrower transition bandwidth, so a trade-off is required between the cost of a small transition bandwidth and the costs of the higher sampling rate, which are those of more storage and higher transmission rates. In many systems the answer has been to make the transition bandwidth between 10 and 20% of the signal band-

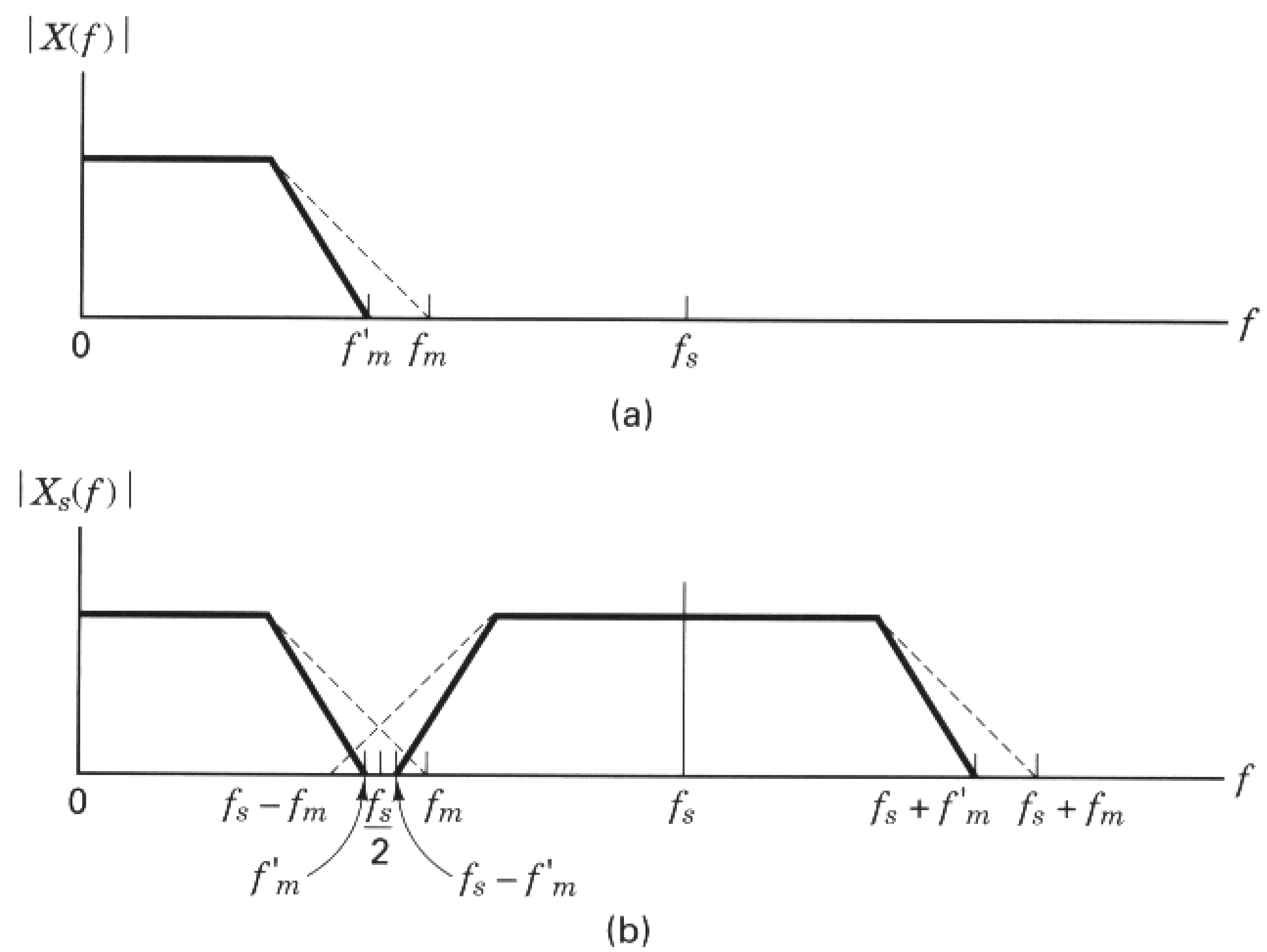


Figure 2.11 Sharper-cutoff filters eliminate aliasing. (a) Continuous signal spectrum. (b) Sampled signal spectrum.

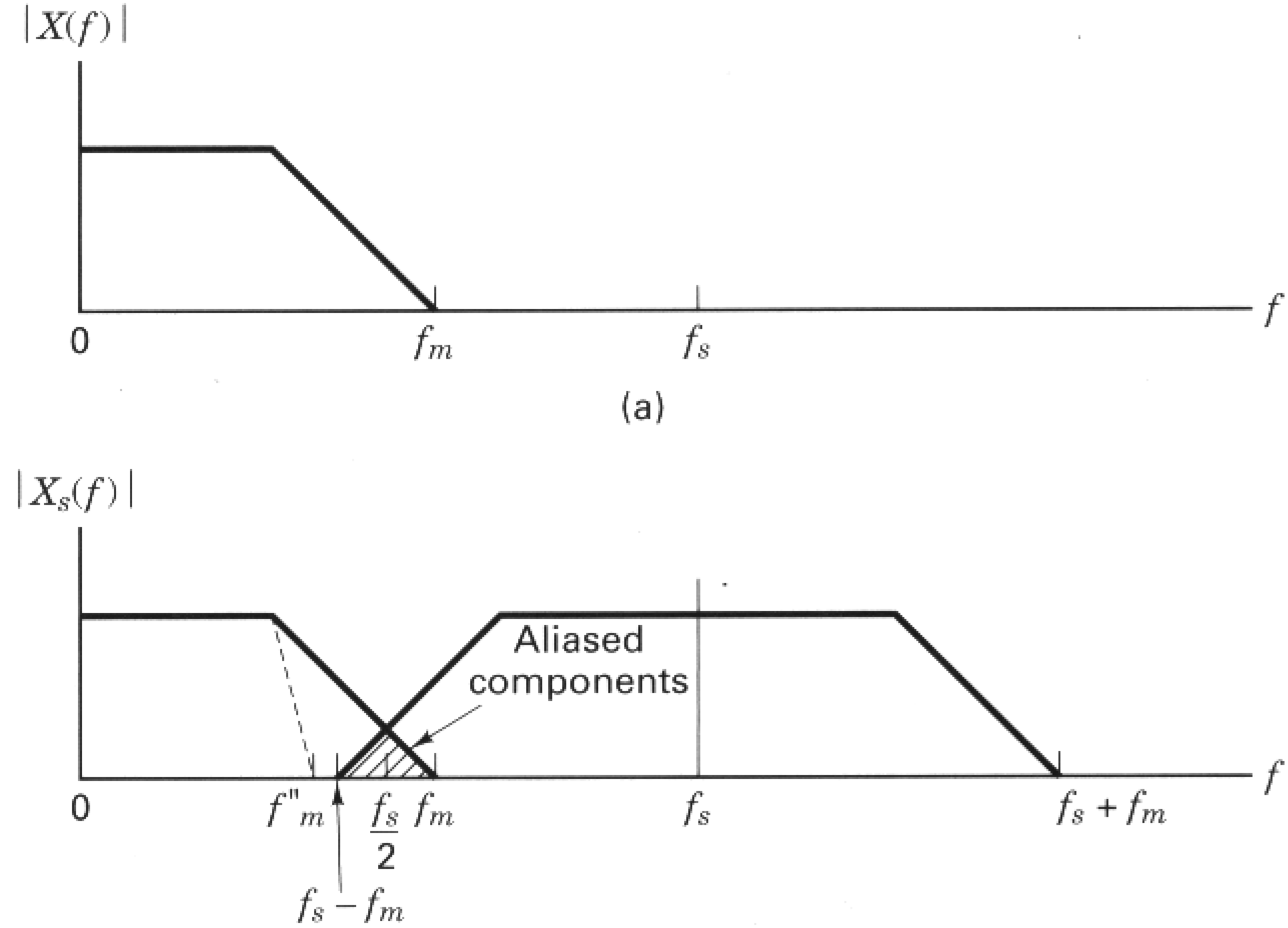


Figure 2.12 Postfilter eliminates aliased portion of spectrum. (a) Continuous signal spectrum. (b) Sampled signal spectrum.

width. If we account for the 20% transition bandwidth of the antialiasing filter, we have an *engineer's version* of the Nyquist sampling rate:

$$f_s \geq 2.2 f_m \tag{2.17}$$

Figure 2.13 provides some insight into aliasing as seen in the time domain. The sampling instants of the solid-line sinusoid have been chosen so that the sinusoidal signal is undersampled. Notice that the resulting ambiguity allows one to draw a totally different (dashed-line) sinusoid, following the undersampled points.

Example 2.2 Sampling Rate for a High-Quality Music System

We wish to produce a high-quality digitization of a 20-kHz bandwidth music source. We are to determine a reasonable sample rate for this source. By the engineer's version of the Nyquist rate, in Equation (2.17), the sampling rate should be greater than 44.0 ksamples/s. As a matter of comparison, the standard sampling rate for the compact disc digital audio player is 44.1 ksamples/s, and the standard sampling rate for studio-quality audio is 48.0 ksamples/s.

2.4.3 Why Oversample?

Oversampling is the most economic solution for the task of transforming an analog signal to a digital signal, or the reverse, transforming a digital signal to an analog signal. This is so because signal processing performed with high performance ana-

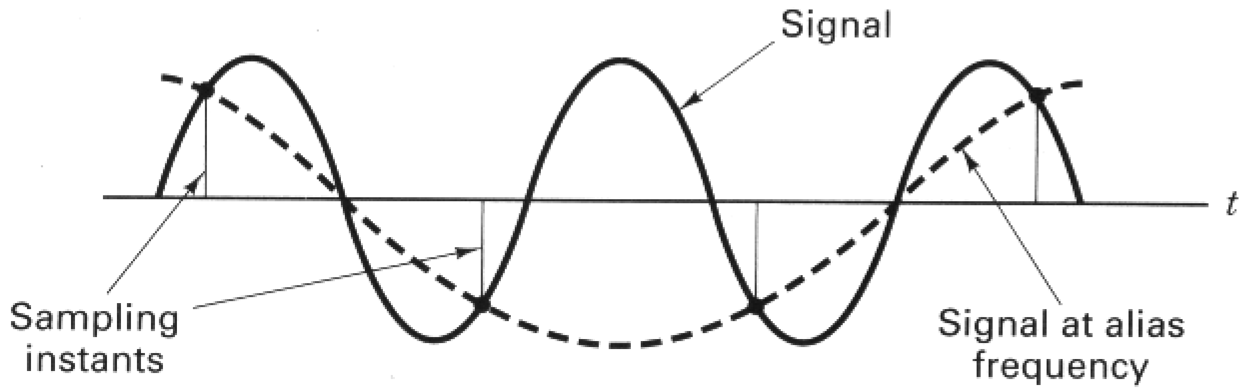


Figure 2.13 Alias frequency generated by sub-Nyquist sampling rate.

log equipment is typically much more costly than using digital signal processing equipment to perform the same task. Consider the task of transforming analog signals to digital signals. When this task is performed without the benefit of over-sampling, the process is characterized by three simple steps, performed in the order that follows.

Without Oversampling

1. The signal passes through a high performance analog lowpass filter to limit its bandwidth.
2. The filtered signal is sampled at the Nyquist rate for the (approximated) bandlimited signal. As described in Section 1.7.2, a strictly bandlimited signal is not realizable.
3. The samples are processed by an analog-to-digital converter that maps the continuous-valued samples to a finite list of discrete output levels.

When this task is performed with the benefit of over-sampling, the process is best described as five simple steps, performed in the order that follows.

With Oversampling

1. The signal is passed through a low performance (less costly) analog low-pass filter (prefilter) to limit its bandwidth.
2. The pre-filtered signal is sampled at the (now higher) Nyquist rate for the (approximated) bandlimited signal.
3. The samples are processed by an analog-to-digital converter that maps the continuous-valued samples to a finite list of discrete output levels.
4. The digital samples are then processed by a high performance digital filter to reduce the bandwidth of the digital samples.
5. The sample rate at the output of the digital filter is reduced in proportion to the bandwidth reduction obtained by this digital filter.

The next two sections examine the benefits of over-sampling.

2.4.3.1 Analog Filtering, Sampling, and Analog to Digital Conversion

The analog filter that limits the bandwidth of an input signal has a passband frequency equal to the signal bandwidth, followed by a transition to a stop band. The bandwidth of the transition region results in an increase in bandwidth of the output signal by some amount f_t. The Nyquist rate f_s for the filtered output, nominally equal to $2f_m$ (twice the highest frequency in the sampled signal) must now be increased to $2f_m + f_t$. The transition bandwidth of the filter represents an overhead in the sampling process. This additional spectral interval does not represent useful signal bandwidth but rather protects the signal bandwidth by reserving a spectral region for the aliased spectrum due to the sampling process. The aliasing stems from the fact that real signals cannot be strictly bandlimited. Typical transition bandwidths represent a 10- to 20-percent increase of the sample rate relative to that dictated by the Nyquist criterion. Examples of this overhead are seen in the compact disc (CD) digital audio system, for which the two-sided bandwidth is 40 kHz and the sample rate is 44.1 kHz, and also in the digital audio tape (DAT) system, which also has a two-sided bandwidth of 40 kHz with a sample rate of 48.0 kHz.

Our intuition and initial impulse is to keep the sample rate as low as possible by building analog filters with narrow transition bandwidths. However, analog filters can exhibit two undesirable characteristics. First, they can exhibit distortion (nonlinear phase versus frequency) due to narrow transition bandwidths. Second, the cost can be high because narrow transition bandwidths dictate high-order filters (see Section 1.6.3.2) requiring a large number of high-quality components. Our quandary is that we wish to operate the sampler at the lowest possible rate to reduce the data-storage cost. To meet this goal we might build a sophisticated analog filter with a narrow transition bandwidth. But such a filter is not only expensive, it also distorts the very signal it has been designed to protect (from undesired aliasing).

The solution (oversampling) is elegant—having been given a problem that we can't solve, we convert it to one that we can solve. We elect to use a low-cost, less sophisticated analog prefilter to limit the bandwidth of the input signal. This analog filter has been simplified by choosing a wider transition bandwidth. With a wider transition bandwidth, the required sample rate must now be increased to accommodate this larger spectrum. We typically start by selecting the higher sample rate to be 4 times the original sample rate, and then we design the analog filter to have a transition bandwidth that matches the increased sample rate. As an example, rather than sampling a CD signal at 44.1 kHz with a transition bandwidth of 4.1 kHz implemented with a sophisticated 10th order elliptic filter (implying that the filter includes 10 energy storage elements, such as capacitors and inductors), we might choose the option to employ oversampling. In that case, we could operate the sampler at 176.4 kHz with a transition bandwidth of 136.4 kHz implemented with a simpler 4th-order elliptic filter (having only 4 energy storage elements).

2.4.3.2 Digital Filtering and Resampling

Now that we have the sampled data, with its higher-than-desired sample rate, we pass the sampled data through a high-performance, low-cost, digital filter to perform the desired anti-alias filtering. The digital filter can realize the narrow

transition bandwidth without the distortion associated with analog filters, and it can operate at low cost. We next reduce the sample rate of the signal (resample) after the digital filtering operation that had reduced the transition bandwidth. Good digital signal processing techniques combine the filtering and the resampling in a single structure.

Now we address a system consideration to further improve the quality of the data collection process. The analog prefilter induces some amplitude and phase distortion. We know precisely what this distortion is, and we design the digital filter so that it not only completes the anti-aliasing task of the analog prefilter, but also compensates for its gain and phase distortion. The composite response can be made as good as we want it to be. Thus we obtain a collected signal of higher quality (less distortion) at reduced cost. Digital signal processing hardware, an extension of the computer industry, is characterized by significantly lower prices each year, which has not been the case with analog processing.

In a similar fashion, oversampling is employed in the process of converting the digital signal to an analog signal (DAC). The analog filter following the DAC suffers from distortion if it has a sharp transition bandwidth. But the transition bandwidth will not be narrow if the output data presented to the DAC has been digitally oversampled.

2.4.4 Signal Interface for a Digital System

Let us examine four ways in which analog source information can be described. Figure 2.14 illustrates the choices. Let us refer to the waveform in Figure 2.14a as the *original analog waveform.* Figure 2.14b represents a sampled version of the original waveform, typically referred to as *natural-sampled data* or PAM (pulse amplitude modulation). Do you suppose that the sampled data in Figure 2.14b are compatible with a digital system? No, they are not, because the amplitude of each natural sample still has an infinite number of possible values; a digital system deals with a finite number of values. Even if the sampling is flat-top sampling, the possible pulse values form an infinite set, since they reflect all the possible values of the continuous analog waveform. Figure 2.14c illustrates the original waveform represented by discrete pulses. Here the pulses have flat tops *and* the pulse amplitude values are limited to a finite set. Each pulse is expressed as a level from a finite number of predetermined levels; each such level can be represented by a symbol from a finite alphabet. The pulses in Figure 2.14c are referred to as *quantized samples;* such a format is the obvious choice for interfacing with a digital system. The format in Figure 2.14d may be construed as the output of a sample-and-hold circuit. When the sample values are quantized to a finite set, this format can also interface with a digital system. After quantization, the analog waveform can still be recovered, but not precisely; improved reconstruction fidelity of the analog waveform can be achieved by increasing the number of quantization levels (requiring increased system bandwidth). Signal distortion due to quantization is treated in the following sections (and later in Chapter 13).

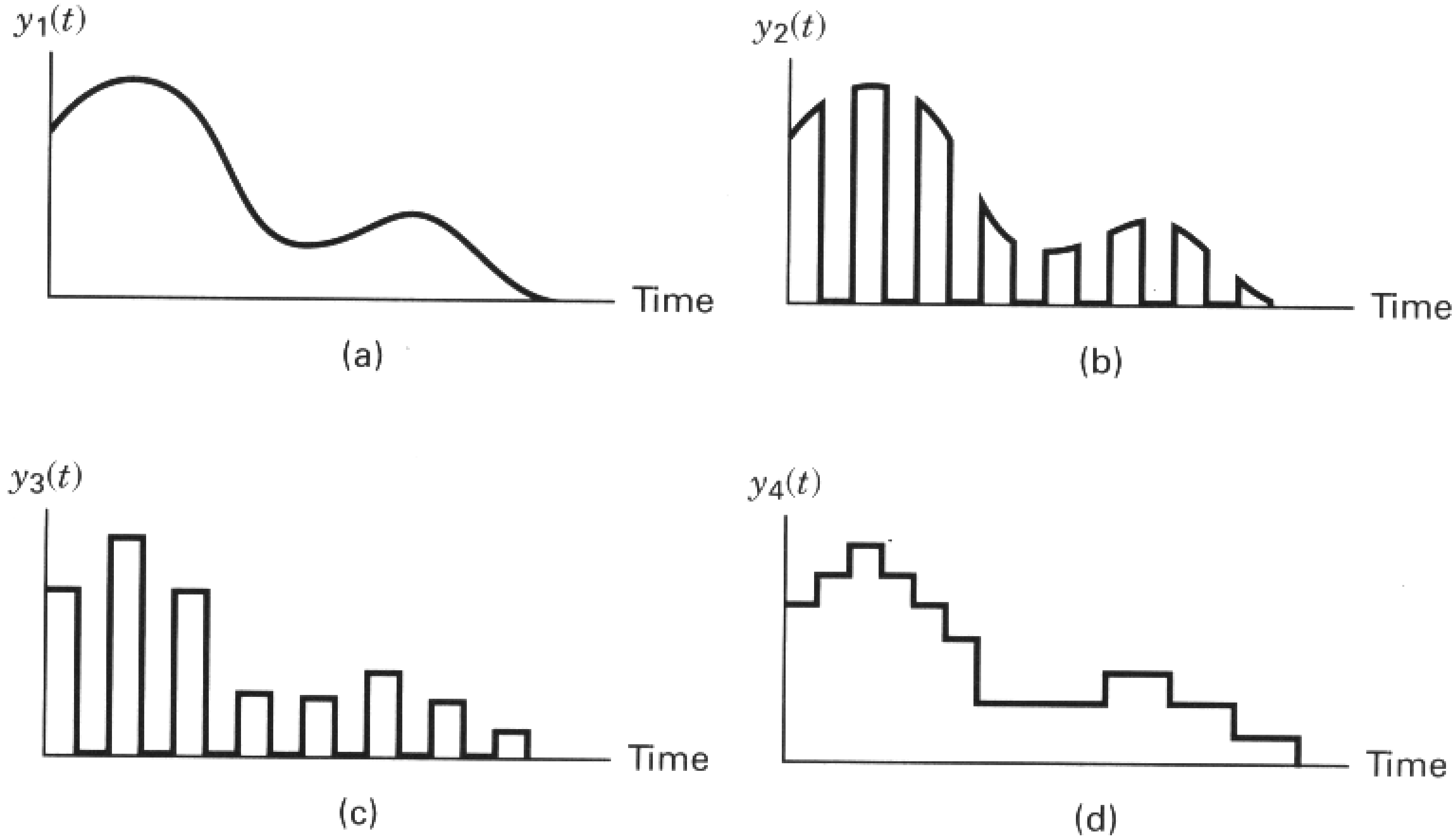


Figure 2.14 Amplitude and time coordinates of source data. (a) Original analog waveform. (b) Natural-sampled data. (c) Quantized samples. (d) Sample and hold.

2.5 SOURCES OF CORRUPTION

The analog signal recovered from the sampled, quantized, and transmitted pulses will contain corruption from several sources. The sources of corruption are related to (1) sampling and quantizing effects, and (2) channel effects. These effects are considered in the sections that follow.

2.5.1 Sampling and Quantizing Effects

2.5.1.1 Quantization Noise

The distortion inherent in quantization is a round-off or truncation error. The process of encoding the PAM signal into a quantized PAM signal involves discarding some of the original analog information. This distortion, introduced by the need to approximate the analog waveform with quantized samples, is referred to as *quantization noise;* the amount of such noise is inversely proportional to the number of levels employed in the quantization process. (The signal-to-noise ratio of quantized pulses is treated in Sections 2.5.3 and 13.2.)

2.5.1.2 Quantizer Saturation

The quantizer (or analog-to-digital converter) allocates L levels to the task of approximating the continuous range of inputs with a finite set of outputs. The range of inputs for which the difference between the input and output is small is

called the *operating range* of the converter. If the input exceeds this range, the difference between the input and the output becomes large, and we say that the converter is operating in *saturation.* Saturation errors, being large, are more objectionable than quantizing noise. Generally, saturation is avoided by the use of automatic gain control (AGC), which effectively extends the operating range of the converter. (Chapter 13 covers quantizer saturation in greater detail.)

2.5.1.3 Timing Jitter

Our analysis of the sampling theorem predicted precise reconstruction of the signal based on uniformly spaced samples of the signal. If there is a slight jitter in the position of the sample, the sampling is no longer uniform. Although exact reconstruction is still possible if the sample positions are accurately known, the jitter is usually a random process and thus the sample positions are not accurately known. The effect of the jitter is equivalent to frequency modulation (FM) of the baseband signal. If the jitter is random, a low-level wideband spectral contribution is induced whose properties are very close to those of the quantizing noise. If the jitter exhibits periodic components, as might be found in data extracted from a tape recorder, the periodic FM will induce low-level spectral lines in the data. Timing jitter can be controlled with very good power supply isolation and stable clock references.

2.5.2 Channel Effects

2.5.2.1 Channel Noise

Thermal noise, interference from other users, and interference from circuit switching transients can cause errors in detecting the pulses carrying the digitized samples. Channel-induced errors can degrade the reconstructed signal quality quite quickly. This rapid degradation of output signal quality with channel-induced errors is called a *threshold effect.* If the channel noise is small, there will be no problem detecting the presence of the waveforms. Thus, small noise does not corrupt the reconstruct signals. In this case, the only noise present in the reconstruction is the quantization noise. On the other hand, if the channel noise is large enough to affect our ability to detect the waveforms, the resulting detection error causes reconstruction errors. A large difference in behavior can occur for very small changes in channel noise level.

2.5.2.2 Intersymbol Interference

The channel is always bandlimited. A bandlimited channel disperses or spreads a pulse waveform passing through it (see Section 1.6.4). When the channel bandwidth is much greater than the pulse bandwidth, the spreading of the pulse will be slight. When the channel bandwidth is close to the signal bandwidth, the spreading will exceed a symbol duration and cause signal pulses to overlap. This overlapping is called *intersymbol interference* (ISI). Like any other source of interference, ISI causes system degradation (higher error rates); it is a particularly

insidious form of interference because raising the signal power to overcome the interference will not always improve the error performance. (Details of how ISI is handled are presented in the next chapter, in Sections 3.3 and 3.4.)

2.5.3 Signal-to-Noise Ratio for Quantized Pulses

Figure 2.15 illustrates an L-level linear quantizer for an analog signal with a peak-to-peak voltage range of $V_{pp} = V_p - (-V_p) = 2V_p$ volts. The quantized pulses assume positive and negative values, as shown in the figure. The step size between quantization levels, called the *quantile interval,* is denoted q volts. When the quantization levels are uniformly distributed over the full range, the quantizer is called a *uniform or linear quantizer.* Each sample value of the analog waveform is approximated with a quantized pulse; the approximation will result in an error no larger than $q/2$ in the positive direction or $-q/2$ in the negative direction. The degradation of the signal due to quantization is therefore limited to half a quantile interval, $\pm q/2$ volts.

A useful figure of merit for the uniform quantizer is the quantizer variance (mean-square error assuming zero mean). If we assume that the quantization error, e, is uniformly distributed over a single quantile interval q-wide (i.e., the analog input takes on all values with equal probability), the quantizer error variance is found to be

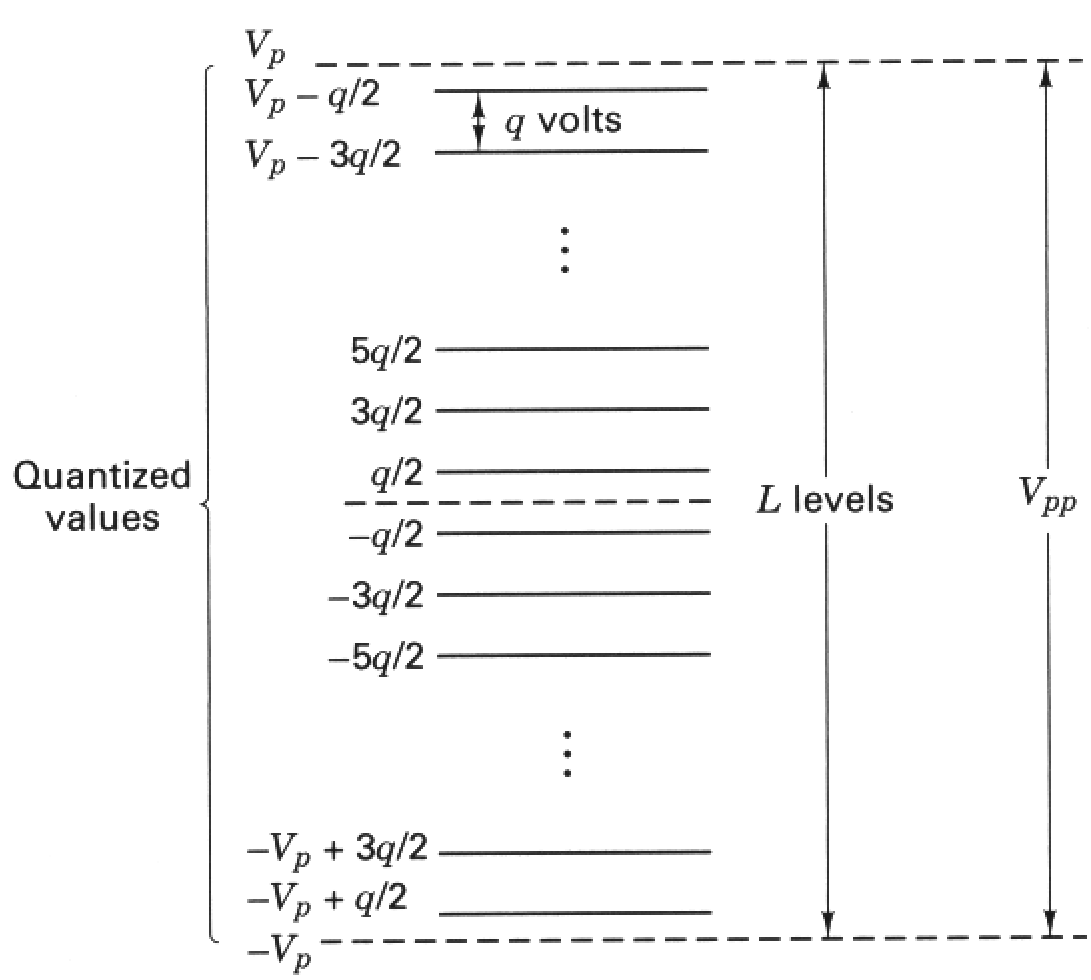


Figure 2.15 Quantization levels.

$$\sigma^2 = \int_{-q/2}^{+q/2} e^2 p(e)\, de \tag{2.18a}$$

$$= \int_{-q/2}^{+q/2} e^2 \frac{1}{q}\, de = \frac{q^2}{12} \tag{2.18b}$$

where $p(e) = 1/q$ is the (uniform) probability density function of the quantization error. The variance, σ^2, corresponds to the *average quantization noise power.* The peak power of the analog signal (normalized to 1 Ω) can be expressed as

$$V_p^2 = \left(\frac{V_{pp}}{2}\right)^2 = \left(\frac{Lq}{2}\right)^2 = \frac{L^2q^2}{4} \tag{2.19}$$

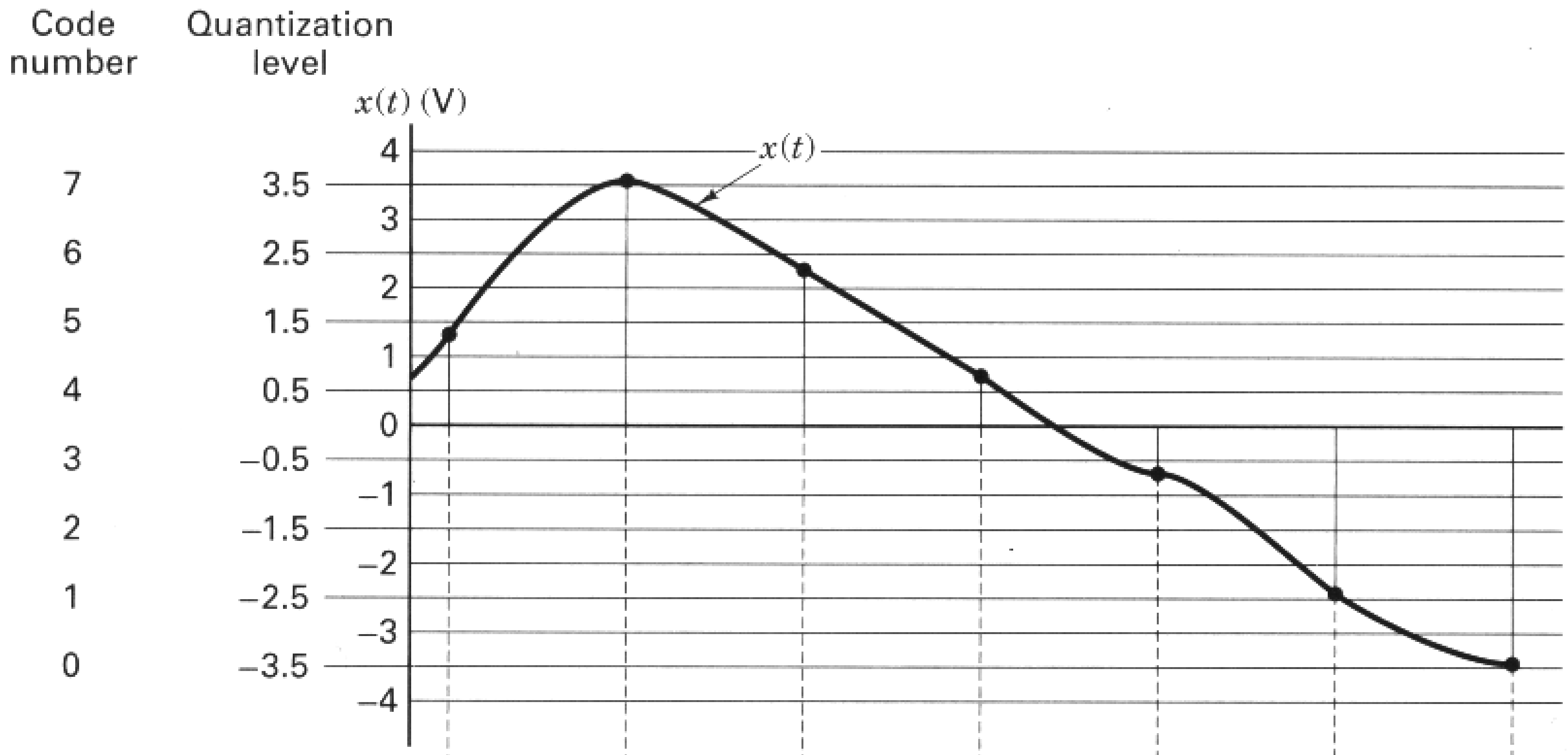
where L is the number of quantization levels. Equations (2.18) and (2.19) combined yield the ratio of *peak* signal power to *average* quantization noise power $(S/N)_q$, assuming that there are no errors due to ISI or channel noise:

$$\left(\frac{S}{N}\right)_q = \frac{L^2q^2/4}{q^2/12} = 3L^2 \tag{2.20}$$

It is intuitively satisfying to see that $(S/N)_q$ improves as a function of the number of quantization levels squared. In the limit (as $L \to \infty$), the signal approaches the PAM format (with no quantization), and the signal-to-quantization noise ratio is infinite; in other words, with an infinite number of quantization levels, there is zero quantization noise.

2.6 PULSE CODE MODULATION

Pulse code modulation (PCM) is the name given to the class of baseband signals obtained from the quantized PAM signals by encoding each quantized sample into a *digital word* [3]. The source information is sampled and quantized to one of L levels; then each quantized sample is digitally encoded into an ℓ-bit ($\ell = \log_2 L$) codeword. For baseband transmission, the codeword bits will then be transformed to pulse waveforms. The essential features of binary PCM are shown in Figure 2.16. Assume that an analog signal $x(t)$ is limited in its excursions to the range –4 to +4 V. The step size between quantization levels has been set at 1 V. Thus, eight quantization levels are employed; these are located at –3.5, –2.5, . . . , +3.5 V. We assign the code number 0 to the level at –3.5 V, the code number 1 to the level at –2.5 V, and so on, until the level at 3.5 V, which is assigned the code number 7. Each code number has its representation in binary arithmetic, ranging from 000 for code number 0 to 111 for code number 7. Why have the voltage levels been chosen in this manner, compared with using a sequence of consecutive integers, 1, 2, 3, . . . ? The choice of voltage levels is guided by two constraints. First, the quantile intervals between the levels should be equal; and second, it is convenient for the levels to be symmetrical about zero.



Natural sample value	1.3	3.6	2.3	0.7	−0.7	−2.4	−3.4
Quantized sample value	1.5	3.5	2.5	0.5	−0.5	−2.5	−3.5
Code number	5	7	6	4	3	1	0
PCM sequence	101	111	110	100	011	001	000

Figure 2.16 Natural samples, quantized samples, and pulse code modulation. (Reprinted with permission from Taub and Schilling, *Principles of Communications Systems,* McGraw-Hill Book Company, New York, 1971, Fig. 6.5-1, p. 205.)

The ordinate in Figure 2.16 is labeled with quantization levels and their code numbers. Each sample of the analog signal is assigned to the quantization level closest to the value of the sample. Beneath the analog waveform $x(t)$ are seen four representations of $x(t)$, as follows: the natural sample values, the quantized sample values, the code numbers, and the PCM sequence.

Note, that in the example of Figure 2.16, each sample is assigned to one of eight levels or a three-bit PCM sequence. Suppose that the analog signal is a musical passage, which is sampled at the Nyquist rate. And, suppose that when we listen to the music in digital form, it sounds terrible. What could we do to improve the fidelity? Recall that the process of quantization replaces the true signal with an approximation (i.e., adds quantization noise). Thus, increasing the number of levels will reduce the quantization noise. If we double the number of levels to 16, what are the consequences? In that case, each analog sample will be represented as a four-bit PCM sequence. Will that cost anything? In a real-time communication system, the messages must not be delayed. Hence, the transmission time for each sample must be the same, regardless of how many bits represent the sample. Hence, when there are more bits per sample, the bits must move faster; in other words, they must be replaced by "skinnier" bits. The data rate is thus increased, and the cost is a greater transmission bandwidth. This explains how one can generally obtain better fidelity at the cost of more transmission bandwidth. Be aware, however,

that there are some communication applications where delay is permissible. For example, consider the transmission of planetary images from a spacecraft. The Galileo project, launched in 1989, was on such a mission to photograph and transmit images of the planet Jupiter. The Galileo spacecraft arrived at its Jupiter destination in 1995. The journey took several years; therefore, any excess signal delay of several minutes (or hours or days) would certainly not be a problem. In such cases, the cost of more quantization levels and greater fidelity need not be bandwidth; it can be time delay.

In Figure 2.1, the term "PCM" appears in two places. First, it is a formatting topic, since the process of analog-to-digital (A/D) conversion involves sampling, quantization, and ultimately yields binary digits via the conversion of quantized PAM to PCM. Here, PCM digits are just binary numbers—a baseband carrier wave has not yet been discussed. The second appearance of PCM in Figure 2.1 is under the heading *Baseband Signaling.* Here, we list various PCM waveforms (line codes) that can be used to "carry" the PCM digits. Therefore, note that the difference between PCM and a PCM waveform is that the former represents a bit sequence, and the latter represents a particular waveform conveyance of that sequence.

2.7 UNIFORM AND NONUNIFORM QUANTIZATION

2.7.1 Statistics of Speech Amplitudes

Speech communication is a very important and specialized area of digital communications. Human speech is characterized by unique statistical properties; one such property is illustrated in Figure 2.17. The abscissa represents speech signal magnitudes, normalized to the root-mean-square (rms) value of such magnitudes through a typical communication channel, and the ordinate is probability. For most voice

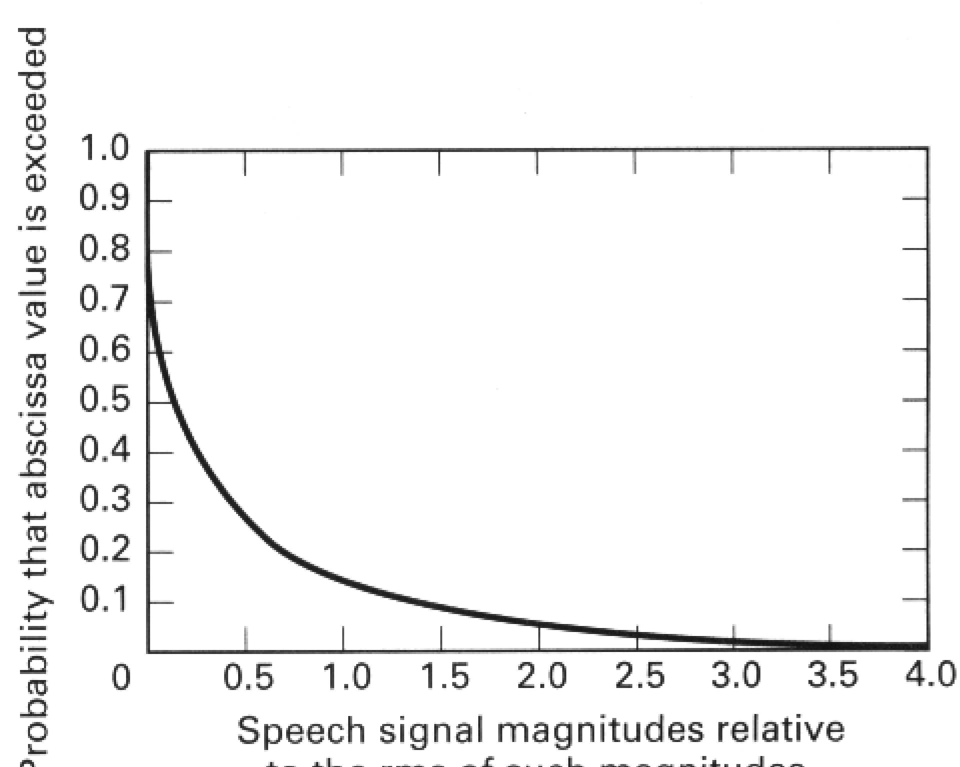


Figure 2.17 Statistical distribution of single-talker speech signal magnitudes.

communication channels, very low speech volumes predominate; 50% of the time, the voltage characterizing detected speech energy is less than one-fourth of the rms value. Large amplitude values are relatively rare; only 15% of the time does the voltage exceed the rms value. We see from Equation (2.18b) that the quantization noise depends on the step size (size of the quantile interval). When the steps are uniform in size the quantization is known as *uniform quantization.* Such a system would be wasteful for speech signals; many of the quantizing steps would rarely be used. In a system that uses equally spaced quantization levels, the quantization noise is the same for all signal magnitudes. Therefore, with uniform quantization, the signal-to-noise (SNR) is worse for low-level signals than for high-level signals. *Nonuniform quantization* can provide fine quantization of the weak signals and coarse quantization of the strong signals. Thus in the case of nonuniform quantization, quantization noise can be made proportional to signal size. The effect is to improve the overall SNR by reducing the noise for the predominant weak signals, at the expense of an increase in noise for the rarely occurring strong signals. Figure 2.18 compares the quantization of a strong versus a weak signal for uniform and nonuniform quantization. The staircase-like waveforms represent the approximations to the analog waveforms (after quantization distortion has been introduced). The SNR improvement that nonuniform quantization provides for the weak signal should be apparent. Nonuniform quantization can be used to make the SNR a constant for all signals within the input range. For voice signals, the typical input signal

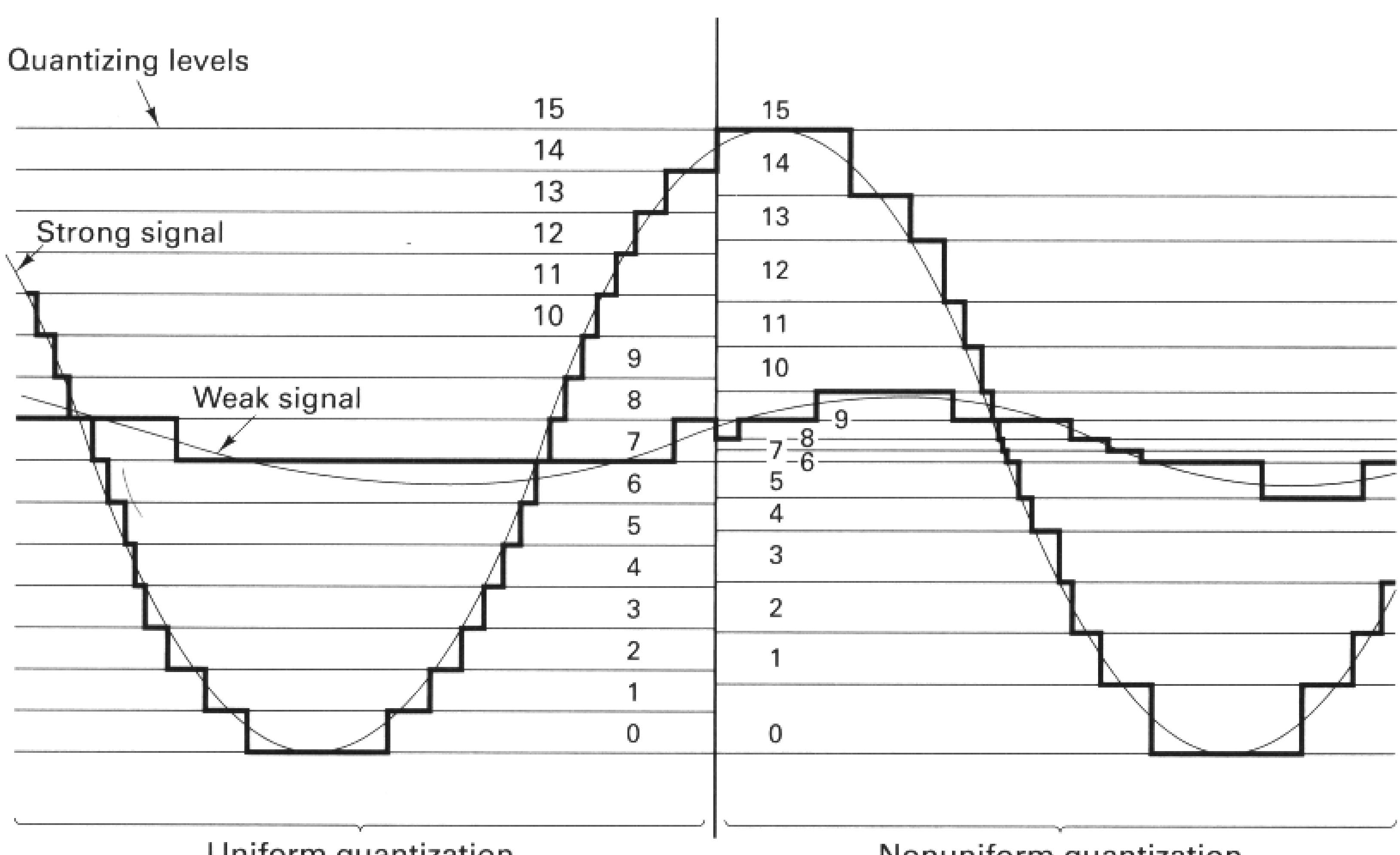


Figure 2.18 Uniform and nonuniform quantization of signals.

dynamic range is 40 decibels (dB), where a decibel is defined in terms of the ratio of power P_2 to power P_1:

$$\text{number of dB} = 10 \log_{10} \frac{P_2}{P_1} \tag{2.21}$$

With a uniform quantizer, weak signals would experience a 40-dB-poorer SNR than that of strong signals. The standard telephone technique of handling the large range of possible input signal levels is to use a *logarithmic-compressed* quantizer instead of a uniform one. With such a nonuniform compressor the output SNR is independent of the distribution of input signal levels.

2.7.2 Nonuniform Quantization

One way of achieving nonuniform quantization is to use a nonuniform quantizer characteristic, shown in Figure 2.19a. More often, nonuniform quantization is achieved by first distorting the original signal with a logarithmic compression characteristic, as shown in Figure 2.19b, and then using a uniform quantizer. For small magnitude signals the compression characteristic has a much steeper slope than for large magnitude signals. Thus, a given signal change at small magnitudes will carry the uniform quantizer through more steps than the same change at large magnitudes. The compression characteristic effectively changes the distribution of the

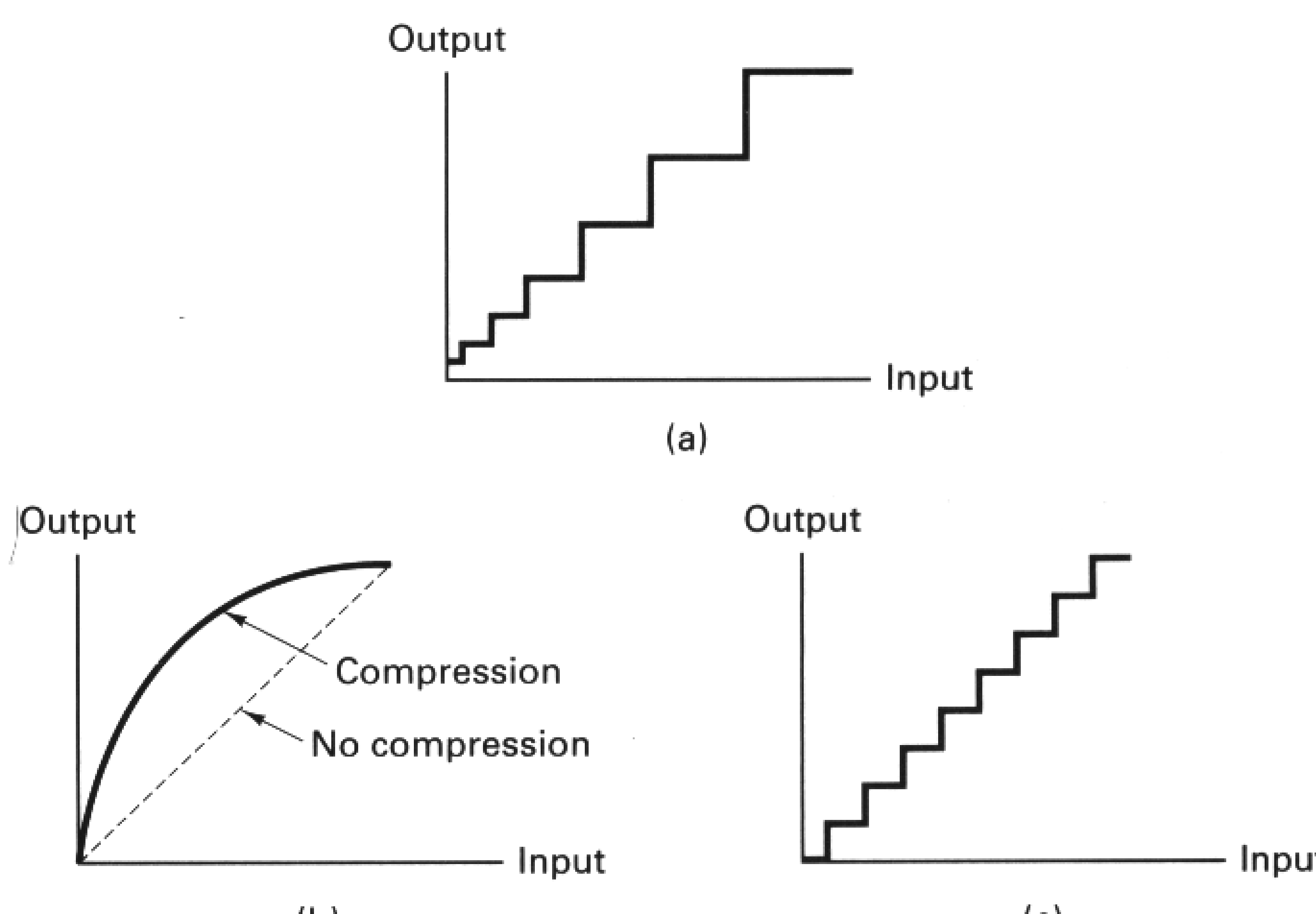


Figure 2.19 (a) Nonuniform quantizer characteristic. (b) Compression characteristic. (c) Uniform quantizer characteristic.

input signal magnitudes so that there is not a preponderance of *low* magnitude signals at the output of the compressor. After compression, the distorted signal is used as the input to a uniform (linear) quantizer characteristic, shown in Figure 2.19c. At the receiver, an inverse compression characteristic, called *expansion,* is applied so that the overall transmission is not distorted. The processing pair (compression and expansion) is usually referred to as *companding.*

2.7.3 Companding Characteristics

The early PCM systems implemented a smooth logarithmic compression function. Today, most PCM systems use a piecewise linear approximation to the logarithmic compression characteristic. In North America, a μ-law compression characteristic

$$y = y_{\max} \frac{\log_e[1 + \mu(|x| / x_{\max})]}{\log_e(1 + \mu)} \operatorname{sgn} x \tag{2.22}$$

is used, where

$$\operatorname{sgn} x = \begin{cases} +1 & \text{for } x \geq 0 \\ -1 & \text{for } x < 0 \end{cases}$$

and where μ is a positive constant, x and y represent input and output voltages, and $x_{\max}$ and $y_{\max}$ are the maximum positive excursions of the input and output voltages, respectively. The compression characteristic is shown in Figure 2.20a for several values of μ. In North America, the standard value for μ is 255. Notice that $\mu = 0$ corresponds to linear amplification (uniform quantization).

Another compression characteristic, used mainly in Europe, is the A-law characteristic, defined as

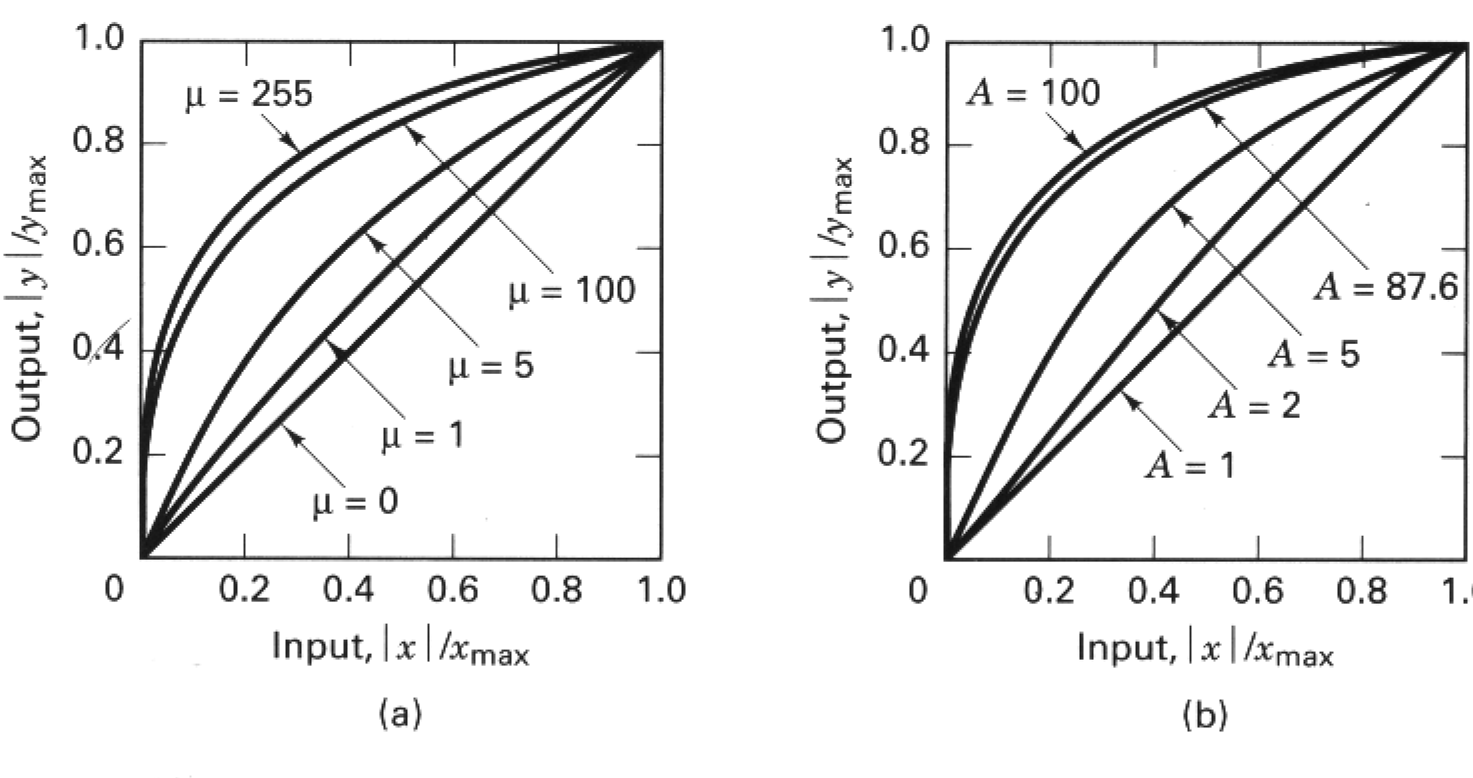


Figure 2.20 Compression characteristics. (a) μ-law characteristic. (b) A-law characteristic.

$$y = \begin{cases} y_{\max} \dfrac{A(|x|/x_{\max})}{1 + \log_e A} \operatorname{sgn} x & 0 < \dfrac{|x|}{x_{\max}} \le \dfrac{1}{A} \\ y_{\max} \dfrac{1 + \log_e [A(|x|/x_{\max})]}{1 + \log_e A} \operatorname{sgn} x & \dfrac{1}{A} < \dfrac{|x|}{x_{\max}} < 1 \end{cases} \tag{2.23}$$

where A is a positive constant and x and y are as defined in Equation (2.22). The A-law compression characteristic is shown in Figure 2.20b for several values of A. A standard value for A is 87.6. (The subjects of uniform and nonuniform quantization are treated further in Chapter 13, Section 13.2.)

2.8 BASEBAND TRANSMISSION

2.8.1 Waveform Representation of Binary Digits

In Section 2.6, it was shown how analog waveforms are transformed into binary digits via the use of PCM. There is nothing "physical" about the digits resulting from this process. Digits are just abstractions—a way to describe the message information. Thus, we need something physical that will represent or "carry" the digits.

We will represent the binary digits with electrical pulses in order to transmit them through a baseband channel. Such a representation is shown in Figure 2.21. Codeword time slots are shown in Figure 2.21a, where the codeword is a 4-bit representation of each quantized sample. In Figure 2.21b, each binary one is represented by a pulse and each binary zero is represented by the absence of a pulse. Thus a sequence of electrical pulses having the pattern shown in Figure 2.21b can be used to transmit the information in the PCM bit stream, and hence the information in the quantized samples of a message.

At the receiver, a determination must be made as to the presence or absence of a pulse in each bit time slot. It will be shown in Section 2.9 that the likelihood of correctly detecting the presence of a pulse is a function of the received pulse energy (or area under the pulse). Thus there is an advantage in making the pulse width T' in Figure 2.21b as wide as possible. If we increase the pulse width to the maximum possible (equal to the bit time T), we have the waveform shown in Figure 2.21c. Rather than describe this waveform as a sequence of present or absent pulses, we can describe it as a sequence of transitions between two levels. When the waveform occupies the upper voltage level it represents a binary one; when it occupies the lower voltage level it represents a binary zero.

2.8.2 PCM Waveform Types

When pulse modulation is applied to a *binary* symbol, the resulting binary waveform is called a pulse-code modulation (PCM) waveform. There are several types of PCM waveforms that are described below and illustrated in Figure 2.22; in telephony applications, these waveforms are often called *line codes.* When pulse modulation is applied to a *nonbinary* symbol, the resulting waveform is called an M-ary

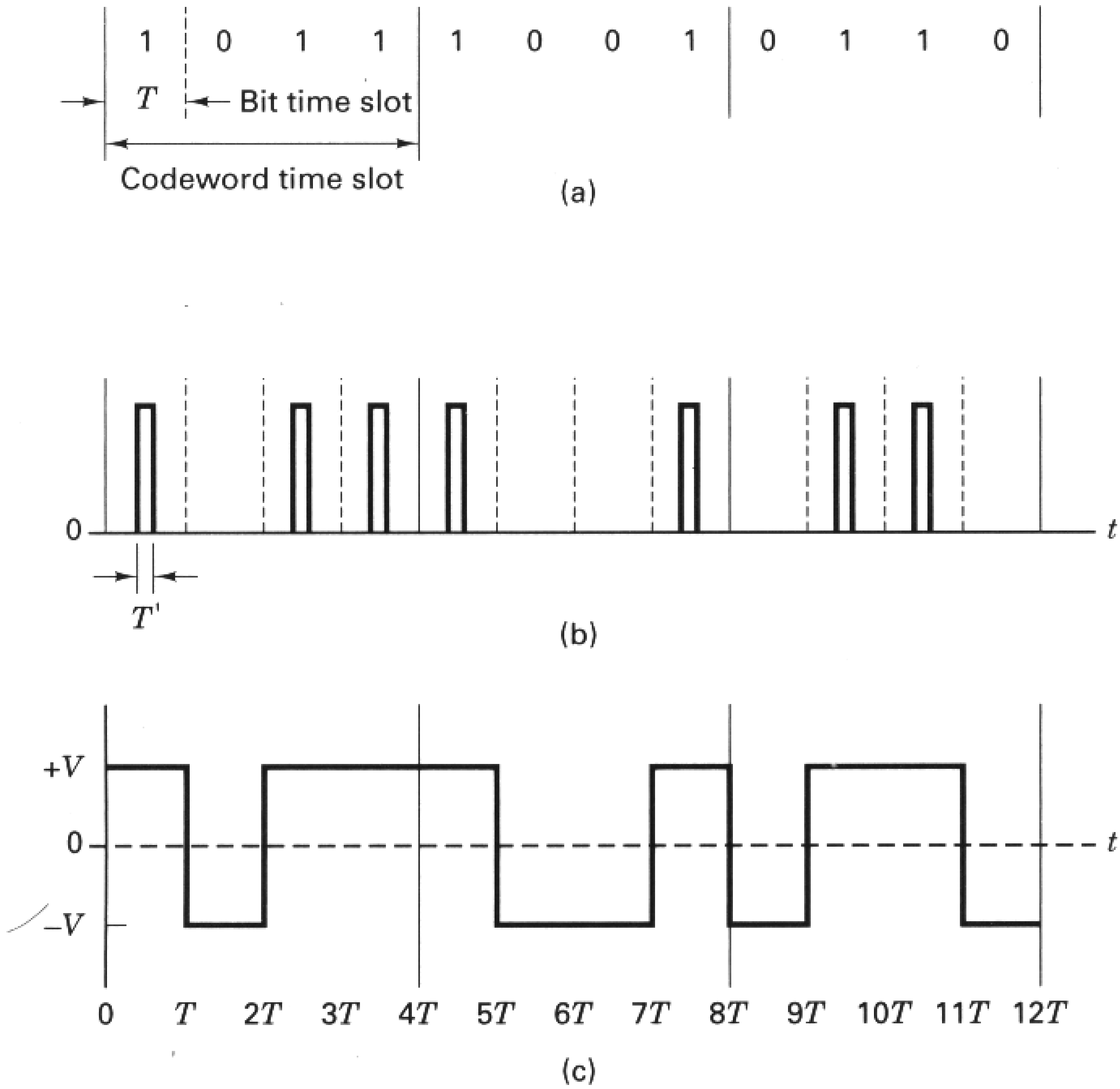


Figure 2.21 Example of waveform representation of binary digits. (a) PCM sequence. (b) Pulse representation of PCM. (c) Pulse waveform (transition between two levels).

pulse-modulation waveform, of which there are several types. They are described in Section 2.8.5, where one of them, called pulse-amplitude modulation (PAM), is emphasized. In Figure 2.1, the highlighted block, labeled *Baseband Signaling,* shows the basic classification of the PCM waveforms and the M-ary pulse waveforms. The PCM waveforms fall into the following four groups.

1. Nonreturn-to-zero (NRZ)
2. Return-to-zero (RZ)
3. Phase encoded
4. Multilevel binary

The NRZ group is probably the most commonly used PCM waveform. It can be partitioned into the following subgroups: NRZ-L (L for level), NRZ-M (M for mark), and NRZ-S (S for space). NRZ-L is used extensively in digital logic circuits.

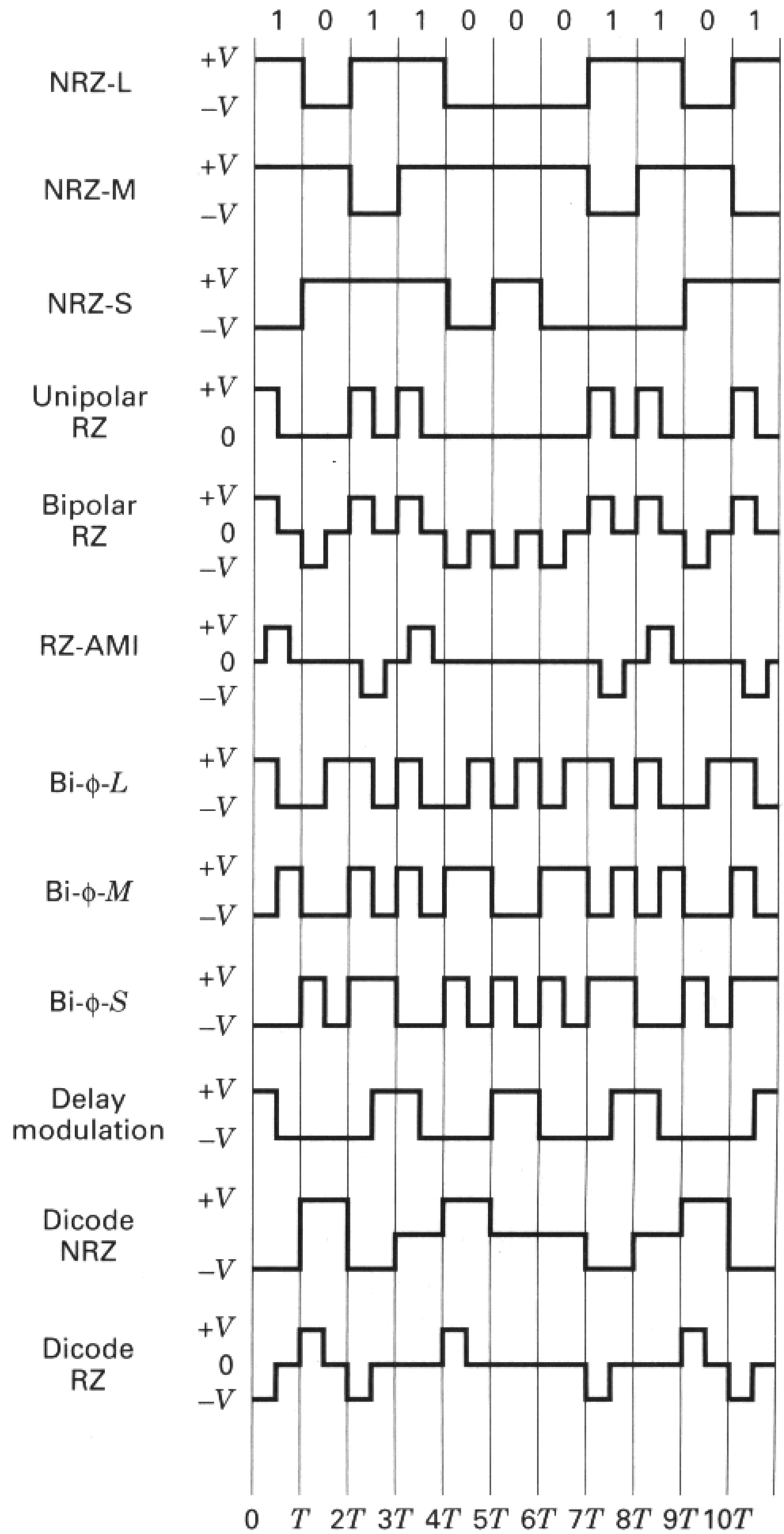


Figure 2.22 Various PCM waveforms.

A binary one is represented by one voltage level and a binary zero is represented by another voltage level. There is a change in level whenever the data change from a one to a zero or from a zero to a one. With NRZ-M, the one, or *mark,* is represented by a change in level, and the zero, or *space,* is represented by no change in level. This is often referred to as *differential encoding.* NRZ-M is used primarily in

magnetic tape recording. NRZ-S is the complement of NRZ-M: A one is represented by no change in level, and a zero is represented by a change in level.

The RZ waveforms consist of unipolar-RZ, bipolar-RZ, and RZ-AMI. These codes find application in baseband data transmission and in magnetic recording. With unipolar-RZ, a one is represented by a half-bit-wide pulse, and a zero is represented by the absence of a pulse. With bipolar-RZ, the ones and zeros are represented by opposite-level pulses that are one-half bit wide. There is a pulse present in each bit interval. RZ-AMI (AMI for "alternate mark inversion") is a signaling scheme used in telephone systems. The ones are represented by equal-amplitude alternating pulses. The zeros are represented by the absence of pulses.

The phase-encoded group consists of bi-ϕ-L (bi-phase-level), better known as *Manchester coding;* bi-ϕ-M (bi-phase-mark); bi-ϕ-S (bi-phase-space); and *delay modulation* (DM), or *Miller coding.* The phase-encoding schemes are used in magnetic recording systems and optical communications and in some satellite telemetry links. With bi-ϕ-L, a one is represented by a half-bit-wide pulse positioned during the first half of the bit interval; a zero is represented by a half-bit-wide pulse positioned during the second half of the bit interval. With bi-ϕ-M, a transition occurs at the beginning of every bit interval. A one is represented by a second transition one-half bit interval later; a zero is represented by no second transition. With bi-ϕ-S, a transition also occurs at the beginning of every bit interval. A one is represented by no second transition; a zero is represented by a second transition one-half bit interval later. With delay modulation [4], a one is represented by a transition at the midpoint of the bit interval. A zero is represented by no transition, unless it is followed by another zero. In this case, a transition is placed at the end of the bit interval of the first zero. Reference to the illustration in Figure 2.22 should help to make these descriptions clear.

Many binary waveforms use three levels, instead of two, to encode the binary data. Bipolar RZ and RZ-AMI belong to this group. The group also contains formats called *dicode* and *duobinary.* With dicode-NRZ, the one-to-zero or zero-to-one data transition changes the pulse polarity; without a data transition, the zero level is sent. With dicode-RZ, the one-to-zero or zero-to-one transition produces a half-duration polarity change; otherwise, a zero level is sent. The three-level duobinary signaling scheme is treated in Section 2.9.

One might ask why there are so many PCM waveforms. Are there really so many unique applications necessitating such a variety of waveforms to represent digits? The reason for the large selection relates to the differences in performance that characterize each waveform [5]. In choosing a PCM waveform for a particular application, some of the parameters worth examining are the following:

1. *Dc component.* Eliminating the dc energy from the signal's power spectrum enables the system to be ac coupled. Magnetic recording systems, or systems using transformer coupling, have little sensitivity to very low frequency signal components. Thus low-frequency information could be lost.
2. *Self-Clocking.* Symbol or bit synchronization is required for any digital communication system. Some PCM coding schemes have inherent synchronizing

or clocking features that aid in the recovery of the clock signal. For example, the Manchester code has a transition in the middle of every bit interval whether a one or a zero is being sent. This guaranteed transition provides a clocking signal.

3. *Error detection.* Some schemes, such as duobinary, provide the means of detecting data errors without introducing additional error-detection bits into the data sequence.
4. *Bandwidth compression.* Some schemes, such as multilevel codes, increase the efficiency of bandwidth utilization by allowing a reduction in required bandwidth for a given data rate; thus there is more information transmitted per unit bandwidth.
5. *Differential encoding.* This technique is useful because it allows the polarity of differentially encoded waveforms to be inverted without affecting the data detection. In communication systems where waveforms sometimes experience inversion, this is a great advantage. (Differential encoding is treated in greater detail in Chapter 4, Section 4.5.2.)
6. *Noise immunity.* The various PCM waveform types can be further characterized by probability of bit error versus signal-to-noise ratio. Some of the schemes are more immune than others to noise. For example, the NRZ waveforms have better error performance than does the unipolar RZ waveform.

2.8.3 Spectral Attributes of PCM Waveforms

The most common criteria used for comparing PCM waveforms and for selecting one waveform type from the many available are spectral characteristics, bit synchronization capabilities, error-detecting capabilities, interference and noise immunity, and cost and complexity of implementation. Figure 2.23 shows the spectral characteristics of some of the most popular PCM waveforms. The figure plots power spectral density in watts/hertz versus normalized bandwidth, WT, where W is bandwidth, and T is the duration of the pulse. WT is often referred to as the *time-bandwidth product,* of the signal. Since the pulse or symbol rate R_s is the reciprocal of T, normalized bandwidth can also be expressed as W/R_s. From this latter expression, we see that the units of normalized bandwidth are hertz/(pulse/s) or hertz/(symbol/s). This is a relative measure of bandwidth; it is valuable because it describes how efficiently the transmission bandwidth is being utilized for each waveform of interest. Any waveform type that requires less than 1.0 Hz for sending 1 symbol/s is relatively bandwidth efficient. Examples would be delay modulation and duobinary (see Section 2.9). By comparison, any waveform type that requires more than 1.0 Hz for sending 1 symbol/s is relatively bandwidth inefficient. An example of this would be bi-phase (Manchester) signaling. From Figure 2.23, we can also see the spectral concentration of signaling energy for each waveform type. For example, NRZ and duobinary schemes have large spectral components at dc and low frequency, while bi-phase has no energy at dc.

An important parameter for measuring *bandwidth efficiency* is R/W having units of bits/s/hz. This measure involves data rate rather than symbol rate. For a

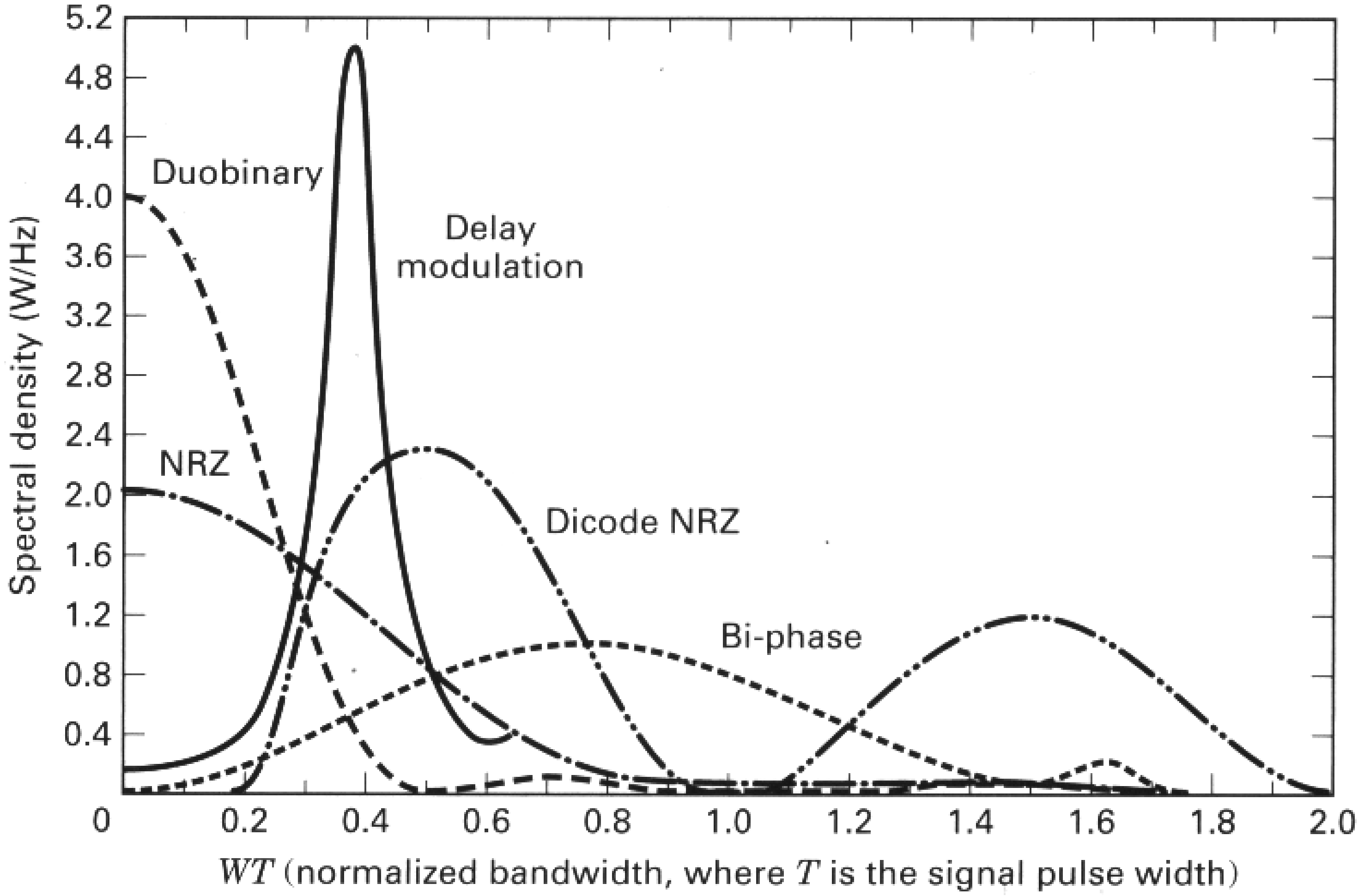


Figure 2.23 Spectral densities of various PCM waveforms.

given signaling scheme, R/W describes how much data throughput can be transmitted for each Hertz of available bandwidth. (Bandwidth efficiency is treated in greater detail in Chapter 9.)

2.8.4 Bits per PCM Word and Bits per Symbol

Throughout Chapters 1 and 2, the idea of binary partitioning ($M = 2^k$) is used to relate the grouping of bits to form symbols for the purpose of signal processing and transmission. We now examine an analogous application where the $M = 2^k$ concept is also applicable. Consider the process of formatting analog information into a bit steam via sampling, quantization, and coding. Each analog sample is transformed into a PCM word made up of groups of bits. The PCM word size can be described by the number of quantization levels allowed for each sample; this is identical to the number of values that the PCM word can assume. Or, the quantization can be described by the number of bits required to identify that set of levels. The relationship between the number of levels per sample and the number of bits needed to represent those levels is the same as the $M = 2^k$ relationship between the size of a set of message symbols and the number of bits needed to represent the symbol. To distinguish between the two applications, the notation is changed for the PCM case. Instead of $M = 2^k$, we use $L = 2^\ell$, where L is the number of quantization levels in the PCM word, and ℓ is the number of bits needed to represent those levels.

2.8.4.1 PCM Word Size

How many bits shall we assign to each analog sample? For digital telephone channels, each speech sample is PCM encoded using 8 bits, yielding 2^8 or 256 levels per sample. The choice of the number of levels, or bits per sample, depends on how much quantization distortion we are willing to tolerate with the PCM format. It is useful to develop a general relationship between the required number of bits per analog sample (the PCM word size), and the allowable quantization distortion. Let the magnitude of the quantization distortion error, $|e|$, be specified as a fraction p of the peak-to-peak analog voltage V_{pp} as follows:

$$|e| \leq pV_{pp} \tag{2.24}$$

Since the quantization error can be no larger than $q/2$, where q is the quantile interval, we can write

$$|e|_{\max} = \frac{q}{2} = \frac{V_{pp}}{2(L-1)} \approx \frac{V_{pp}}{2L} \tag{2.25}$$

where L is the number of quantization levels. For most applications the number of levels is large enough so that $L - 1$ can be replaced by L, as was done above. Then, from Equations (2.24) and (2.25), we can write

$$\frac{V_{pp}}{2L} \leq pV_{pp} \tag{2.26}$$

$$2^{\ell} = L \geq \frac{1}{2p} \quad \text{levels} \tag{2.27}$$

and

$$\ell \geq \log_2 \frac{1}{2p} \quad \text{bits} \tag{2.28}$$

It is important that we do not confuse the idea of bits per PCM word, denoted by ℓ in Equation (2.28), with the M-level transmission concept of k data bits per symbol. (Example 2.3, presented shortly, should clarify the distinction.)

2.8.5 *M*-ary Pulse-Modulation Waveforms

There are three basic ways to modulate information on to a sequence of pulses: we can vary the pulse's amplitude, position, or duration, which leads to the names *pulse-amplitude modulation* (PAM), *pulse-position modulation* (PPM), and *pulse-duration modulation* (PDM), respectively. PDM is sometimes called pulse-width modulation (PWM). When information samples without any quantization are modulated on to pulses, the resulting pulse modulation can be called *analog pulse modulation.* When the information samples are first quantized, yielding symbols from an M-ary alphabet set, and then modulated on to pulses, the resulting pulse modulation is digital and we refer to it as *M-ary pulse modulation.* In the case of M-ary PAM, one of M allowable amplitude levels are assigned to each of the M possible symbol values. Earlier we described PCM waveforms as binary waveforms having

two amplitude values (e.g., NRZ, RZ). Note that such PCM waveforms requiring only two levels represent the special case ($M = 2$) of the general M-ary PAM that requires M levels. In this book, the PCM waveforms are grouped separately (see Figure 2.1 and Section 2.8.2) and are emphasized because they are the most popular of the pulse-modulation schemes.

In the case of M-ary PPM waveforms, modulation is effected by delaying (or advancing) a pulse occurrence, by an amount that corresponds to the value of the information symbols. In the case of M-ary PDM waveforms, modulation is effected by varying the pulse width by an amount that corresponds to the value of the symbols. For both PPM and PDM, the pulse amplitude is held constant. Baseband modulation with pulses have analogous counterparts in the area of bandpass modulation. PAM is similar to amplitude modulation, while PPM and PDM are similar to phase and frequency modulation respectively. In this section, we only address M-ary PAM waveforms as they compare to PCM waveforms.

The transmission bandwidth required for binary digital waveforms such as PCM may be very large. What might we do to reduce the required bandwidth? One possibility is to use *multilevel signaling.* Consider a bit stream with data rate, R bits per second. Instead of transmitting a pulse waveform for each bit, we might first partition the data into k-bit groups, and then use ($M = 2^k$)-level pulses for transmission. With such multilevel signaling or M-ary PAM, each pulse waveform can now represent a k-bit symbol in a symbol stream moving at the rate of R/k symbols per second (a factor k slower than the bit stream). Thus for a given data rate, multilevel signaling, where $M > 2$, can be used to reduce the number of symbols transmitted per second; or, in other words, M-ary PAM as opposed to binary PCM can be used to reduce the transmission bandwidth requirements of the channel. Is there a price to be paid for such bandwidth reduction? Of course, and that is discussed below.

Consider the task that the pulse receiver must perform: It must distinguish between the possible levels of each pulse. Can the receiver distinguish among the eight possible levels of each octal pulse in Figure 2.24a as easily as it can distinguish between the two possible levels of each binary pulse in Figure 2.24b? The transmission of an 8-level (compared with a 2-level) pulse requires a greater amount of energy for equivalent detection performance. (It is the amount of received E_b/N_0 that determines how reliably a signal will be detected). For equal average power in the binary and the octal pulses, it is easier to detect the binary pulses because the detector has more signal energy per level for making a binary decision that an 8-level decision. What price does a system designer pay if he or she chooses the transmission waveform to be the easier-to-detect binary PCM rather than the 8-level PAM? The engineer pays the price of needing three times as much transmission bandwidth for a given data rate, compared with the octal pulses, since each octal pulse must be replaced with three binary pulses (each one-third as wide as the octal pulses). One might ask, Why not use binary pulses with the same pulse duration as the original octal pulses and suffer the information delay? For some cases, this might be appropriate, but for real-time communication systems, such an increase in delay cannot be tolerated—the 6 o'clock news *must* be received at 6 o-clock. (In Chapter 9, we examine in detail the trade-off between signal power and transmission bandwidth.)

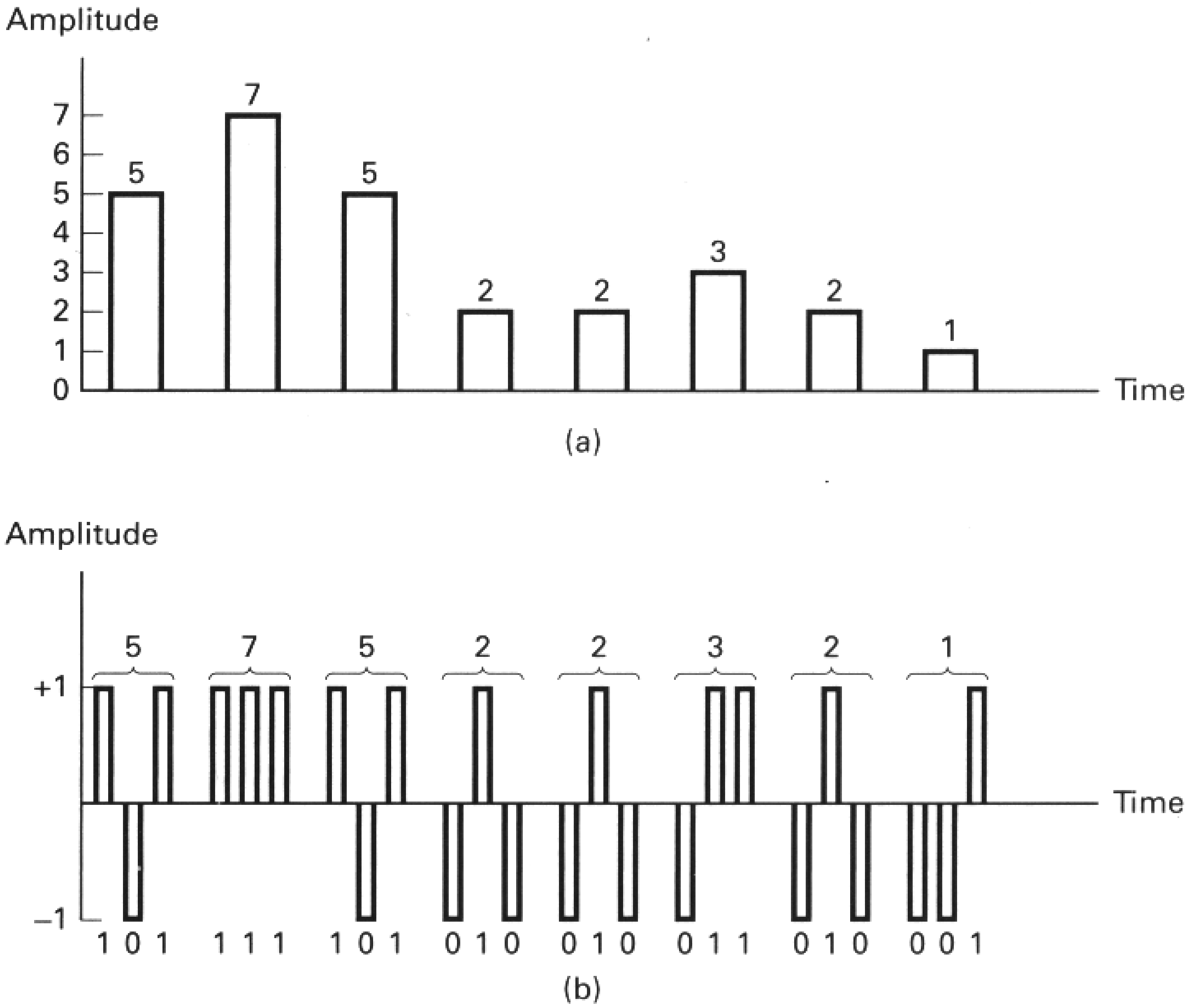


Figure 2.24 Pulse code modulation signaling. (a) Eight-level signaling. (b) Two-level signaling.

Example 2.3 Quantization Levels and Multilevel Signaling

The information in an analog waveform, with maximum frequency $f_m = 3$ kHz, is to be transmitted over an M-ary PAM system, where the number of pulse levels is $M = 16$. The quantization distortion is specified not to exceed $\pm 1\%$ of the peak-to-peak analog signal.

(a) What is the minimum number of bits/sample, or bits/PCM word that should be used in digitizing the analog waveform?
(b) What is the minimum required sampling rate, and what is the resulting bit transmission rate?
(c) What is the PAM pulse or symbol transmission rate?
(d) If the transmission bandwidth (including filtering) equals 12 kHz, determine the bandwidth efficiency for this system.

In this example we are concerned with two types of *levels:* the number of quantization levels for fulfilling the distortion requirement and the 16 levels of the multilevel PAM pulses.

Solution

(a) Using Equation (2.28), we calculate

$$\ell \geq \log_2 \frac{1}{0.02} = \log_2 50 \approx 5.6.$$

Therefore, use $\ell = 6$ bits/sample to meet the distortion requirement.

(b) Using the Nyquist sampling criterion, the minimum sampling rate $f_s = 2f_m = 6000$ samples/second. From part (a), each sample will give rise to a PCM word composed of 6 bits. Therefore the bit transmission rare $R = \ell f_s = 36{,}000$ bits/sec.

(c) Since multilevel pulses are to be used with $M = 2^k = 16$ levels, then $k = \log_2 16 = 4$ bits/symbol. Therefore, the bit stream will be partitioned into groups of 4 bits to form the new 16-level PAM digits, and the resulting symbol transmission rate R_s is $R/k = 36{,}000/4 = 9000$ symbols/s.

(d) Bandwidth efficiency is described by data throughput per hertz, R/W. Since $R = 36{,}000$ bits/s, and $W = 12$ kHz, then $R/W = 3$ bits/s/hz.

2.9 CORRELATIVE CODING

In 1963, Adam Lender [6, 7] showed that it is possible to transmit $2W$ symbols/s with zero ISI, using the theoretical minimum bandwidth of W hertz, without infinitely sharp filters. Lender used a technique called *duobinary signaling,* also referred to as *correlative coding* and *partial response signaling.* The basic idea behind the duobinary technique is to introduce some controlled amount of ISI into the data stream rather than trying to eliminate it completely. By introducing correlated interference between the pulses, and by changing the detection procedure, Lender, in effect, "canceled out" the interference at the detector and thereby achieved the ideal symbol-rate packing of 2 symbols/s/Hz, an amount that had been considered unrealizable.

2.9.1 Duobinary Signaling

To understand how duobinary signaling introduces controlled ISI, let us look at a model of the process. We can think of the duobinary coding operation as if it were implemented as shown in Figure 2.25. Assume that a sequence of binary symbols $\{x_k\}$ is to be transmitted at the rate of R symbols/s over a system having an ideal rectangular spectrum of bandwidth $W = R/2 = 1/2T$ hertz. You might ask: How is this rectangular spectrum, in Figure 2.25, different from the unrealizable Nyquist characteristic? It has the same ideal characteristic; but we are not trying to implement the ideal rectangular filter. It is only the part of our equivalent model that is used for developing a filter that is easier to approximate. Before being shaped by the ideal filter, the pulses pass through a simple digital filter, as shown in the figure. The digital filter incorporates a one-digit delay; to each incoming pulse, the filter adds the value of the previous pulse. In other words, for every pulse into the digital filter, we get the summation of two pulses out. Each pulse of the sequence $\{y_k\}$ out of the digital filter can be expressed as

$$y_k = x_k + x_{k-1} \tag{2.29}$$

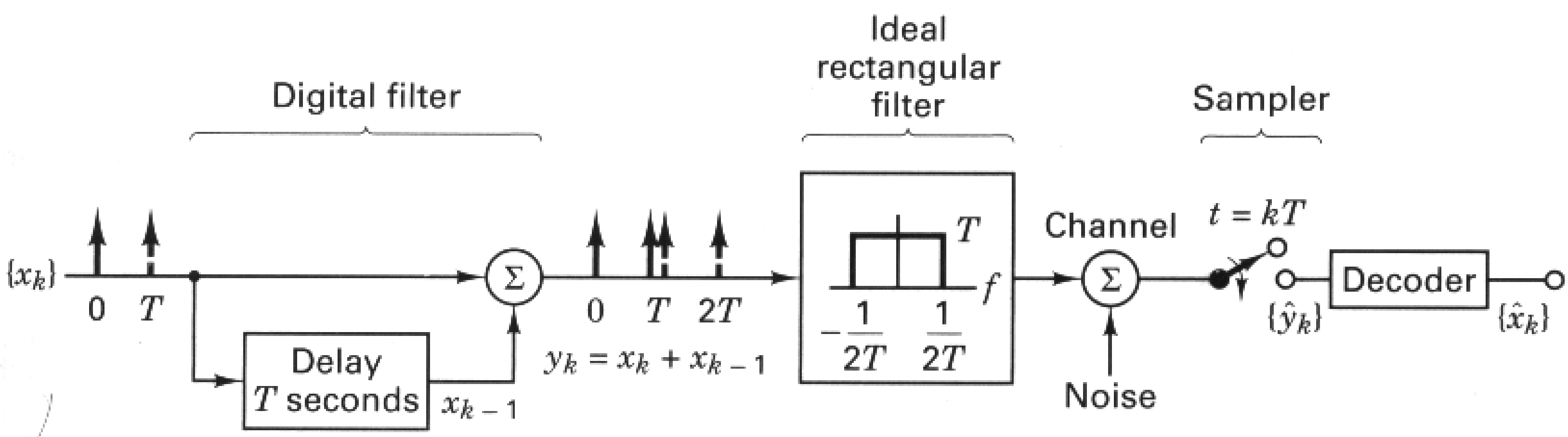


Figure 2.25 Duobinary signaling.

Hence, the $\{y_k\}$ amplitudes are not independent; each y_k digit carries with it the *memory* of the prior digit. The ISI introduced to each y_k digit comes only from the preceding x_{k-1} digit. This correlation between the pulse amplitudes of $\{y_k\}$ can be thought of as the controlled ISI introduced by the duobinary coding. Controlled interference is the essence of this novel technique because at the detector, such controlled interference can be removed as easily as it was added. The $\{y_k\}$ sequence is followed by the ideal Nyquist filter that does not introduce any ISI. In Figure 2.25, at the receiver sampler, we would expect to recover the sequence $\{y_k\}$ exactly in the absence of noise. Since all systems experience noise contamination, we shall refer to the *received* $\{y_k\}$ as the estimate of $\{y_k\}$ and denote it $\{\hat{y}_k\}$. Removing the controlled interference with the duobinary decoder yields an estimate of $\{x_k\}$ which we shall denote as $\{\hat{x}_k\}$.

2.9.2 Duobinary Decoding

If the binary digit x_k is equal to ± 1, then using Equation (2.29), y_k has one of three possible values: $+2$, 0, or -2. The duobinary code results in a three-level output: in general, for M-ary transmission, partial response signaling results in $2M - 1$ output levels. The decoding procedure involves the inverse of the coding procedure, namely, subtracting the x_{k-1} decision from the y_k digit. Consider the following coding/decoding example.

Example 2.4 Duobinary Coding and Decoding

Use Equation (2.29) to demonstrate duobinary coding and decoding for the following sequence: $\{x_k\} = 0\ 0\ 1\ 0\ 1\ 1\ 0$. Consider the first bit of the sequence to be a startup digit, not part of the data.

Solution

Binary digit sequence $\{x_k\}$:	0	0	1	0	1	1	0
Bipolar amplitudes $\{x_k\}$:	−1	−1	+1	−1	+1	+1	−1
Coding rule: $y_k = x_k + x_{k-1}$:		−2	0	0	0	2	0

Decoding decision rule:	If $\hat{y}_k = 2$, decide that $\hat{x}_k = +1$ (or binary one).					
	If $\hat{y}_k = -2$, decide that $\hat{x}_k = -1$ (or binary zero).					
	If $\hat{y}_k = 0$, decide opposite of the previous decision.					
Decoded bipolar sequence $\{\hat{x}_k\}$:	−1	+1	−1	+1	+1	−1
Decoded binary sequence $\{\hat{x}_k\}$:	0	1	0	1	1	0

The decision rule simply implements the subtraction of each $\hat{x}_{k-1}$ decision from each $\hat{y}_k$. One drawback of this detection technique is that once an error is made, it tends to propagate, causing further errors, since present decisions depend on prior decisions. A means of avoiding this error propagation is known as *precoding.*

2.9.3 Precoding

Precoding is accomplished by first differentially encoding the $\{x_k\}$ binary sequence into a new $\{w_k\}$ binary sequence by means of the equation:

$$w_k = x_k \oplus w_{k-1} \tag{2.30}$$

where the symbol $\oplus$ represents modulo-2 addition (equivalent to the logical *exclusive-or* operation) of the binary digits. The rules of modulo-2 addition are as follows:

$$\begin{aligned} 0 \oplus 0 &= 0 \\ 0 \oplus 1 &= 1 \\ 1 \oplus 0 &= 1 \\ 1 \oplus 1 &= 0 \end{aligned}$$

The $\{w_k\}$ binary sequence is then converted to a bipolar pulse sequence, and the coding operation proceeds in the same way as it did in Example 2.4. However, with precoding, the detection process is quite different from the detection of ordinary duobinary, as shown below in Example 2.5: The precoding model is shown in Figure 2.26; in this figure it is implicit that the modulo-2 addition producing the precoded $\{w_k\}$ sequence is performed on the *binary* digits, while the digital filtering producing the $\{y_k\}$ sequence is performed on the *bipolar* pulses.

Example 2.5 Duobinary Precoding

Illustrate the duobinary coding and decoding rules when using the differential precoding of Equation (2.30). Assume the same $\{x_k\}$ sequence as that given in Example 2.4.

Solution

Binary digit sequence $\{x_k\}$:	0	0	1	0	1	1	0
Precoded sequence $w_k = x_k \oplus w_{k-1}$:	0	0	1	1	0	1	1
Bipolar sequence $\{w_k\}$:	−1	−1	+1	+1	−1	+1	+1
Coding rule: $y_k = w_k + w_{k-1}$:		−2	0	+2	0	0	+2

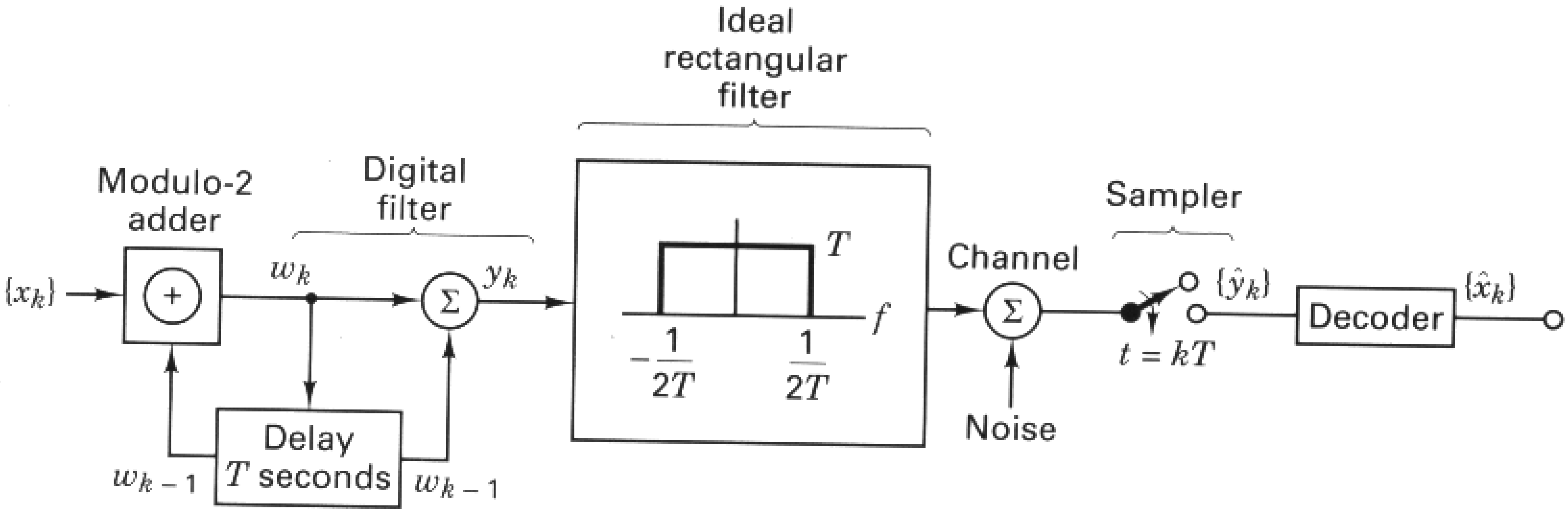


Figure 2.26 Precoded duobinary signaling.

Decoding decision rule: If $\hat{y}_k = \pm 2$, decide that $\hat{x}_k$ = binary zero.
If $\hat{y}_k = 0$, decide that $\hat{x}_k$ = binary one.

Decoded binary sequence $\{x_k\}$: 0 1 0 1 1 0

The differential precoding enables us to decode the $\{\hat{y}_k\}$ sequence by making a decision on each received sample singly, without resorting to prior decisions that could be in error. The major advantage is that in the event of a digit error due to noise, such an error does not propagate to other digits. Notice that the first bit in the differentially precoded binary sequence $\{w_k\}$ is an arbitrary choice. If the startup bit in $\{w_k\}$ had been chosen to be a binary one instead of a binary zero, the decoded result would have been the same.

2.9.4 Duobinary Equivalent Transfer Function

In Section 2.9.1, we described the duobinary transfer function as a digital filter incorporating a one-digit delay followed by an ideal rectangular transfer function. Let us now examine an equivalent model. The Fourier transform of a delay can be described as $e^{-j2\pi fT}$ (see Section A.3.1); therefore, the input digital filter of Figure 2.25 can be characterized as the frequency transfer function

$$H_1(f) = 1 + e^{-j2\pi fT} \tag{2.31}$$

The transfer function of the ideal rectangular filter, is

$$H_2(f) = \begin{cases} T & \text{for } |f| < \dfrac{1}{2T} \\ 0 & \text{elsewhere} \end{cases} \tag{2.32}$$

The overall equivalent transfer function of the digital filter cascaded with the ideal rectangular filter is then given by

$$
\begin{aligned}
H_e(f) &= H_1(f)H_2(f) \qquad \text{for } |f| < \frac{1}{2T} \\
&= (1 + e^{-j2\pi fT})T \\
&= T(e^{j\pi fT} + e^{-j\pi fT})e^{-j\pi fT}
\end{aligned}
\tag{2.33}
$$

so that

$$
|H_e(f)| = \begin{cases} 2T\cos \pi fT & \text{for } |f| < \dfrac{1}{2T} \\ 0 & \text{elsewhere} \end{cases}
\tag{2.34}
$$

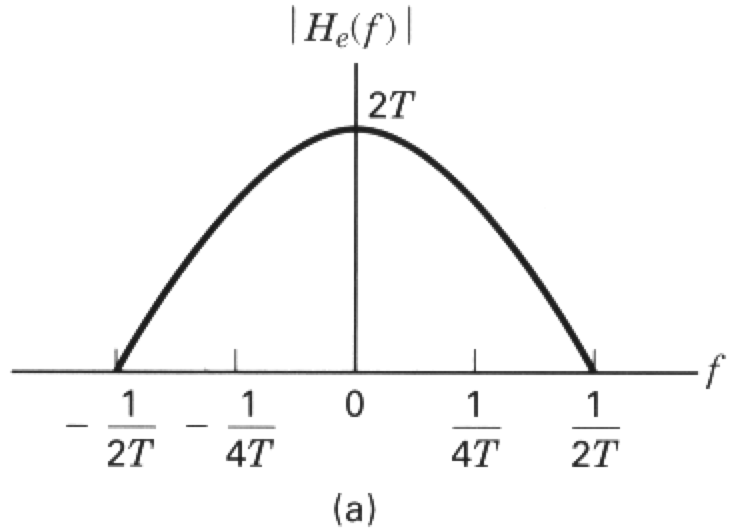
Thus $H_e(f)$, the composite transfer function for the cascaded digital and rectangular filters, has a gradual roll-off to the band edge, as can be seen in Figure 2.27a. The transfer function can be approximated by using realizable analog filtering; a separate digital filter is not needed. The duobinary equivalent $H_e(f)$ is called a *cosine filter* [8]. The cosine filter should not be confused with the *raised* cosine filter (described in Chapter 3, Section 3.3.1). The corresponding impulse response $h_e(t)$, found by taking the inverse Fourier transform of $H_e(f)$ in Equation (2.33) is

$$
h_e(t) = \operatorname{sinc}\left(\frac{t}{T}\right) + \operatorname{sinc}\left(\frac{t-T}{T}\right)
\tag{2.35}
$$

and is plotted in Figure 2.27b. For every impulse $\delta(t)$ at the input of Figure 2.25, the output is $h_e(t)$ with an appropriate polarity. Notice that there are only two nonzero samples at T-second intervals, giving rise to controlled ISI from the adjacent bit. The introduced ISI is eliminated by use of the decoding procedure discussed in Section 2.9.2. Although the cosine filter is noncausal and therefore nonrealizable, it can be easily approximated. The implementation of the precoded duobinary technique described in Section 2.9.3 can be accomplished by first differentially encoding the binary sequence $\{x_k\}$ into the sequence $\{w_k\}$ (see Example 2.5). The pulse sequence $\{w_k\}$ is then filtered by the equivalent cosine characteristic described in Equation (2.34).

2.9.5 Comparison of Binary with Duobinary Signaling

The duobinary technique introduces correlation between pulse amplitudes, whereas the more restrictive Nyquist criterion assumes that the transmitted pulse amplitudes are independent of one another. We have shown that duobinary signaling can exploit this introduced correlation to achieve zero ISI signal transmission, using a smaller system bandwidth than is otherwise possible. Do we get this performance improvement without paying a price? No, such is rarely the case with engineering design options—there is almost always a trade-off involved. We saw that duobinary coding requires three levels, compared with the usual two levels for binary coding. Recall our discussion in Section 2.8.5, where we compared the performance and the required signal power for making eight-level PAM decisions versus two-level (PCM) decisions. For a fixed amount of signal power, the ease of making reliable decisions is inversely related to the number of levels that must be



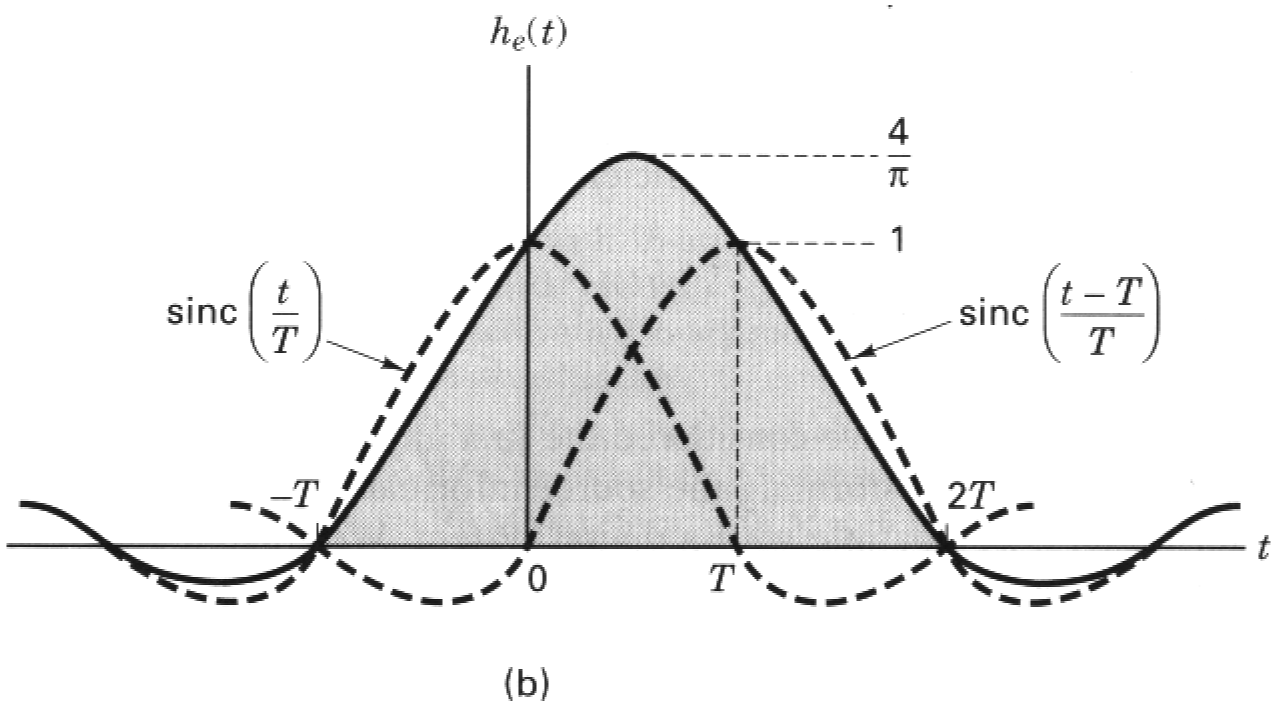


Figure 2.27 Duobinary transfer function and pulse shape. (a) Cosine filter. (b) Impulse response of the cosine filter.

distinguished in each waveform. Therefore, it should be no surprise that although duobinary signaling accomplishes the zero ISI requirement with minimum bandwidth, duobinary signaling also requires more power than binary signaling, for equivalent performance against noise. For a given probability of bit error (P_B), duobinary signaling requires approximately 2.5 dB greater SNR than binary signaling, while using only $1/(1 + r)$ the bandwidth that binary signaling requires [7], where r is the filter roll-off.

2.9.6 Polybinary Signaling

Duobinary signaling can be extended to more than three digits or levels, resulting in greater bandwidth efficiency; such systems are called *polybinary* [7, 9]. Consider that a binary message with two signaling levels is transformed into a signal with j signaling levels numbered consecutively from zero to $(j - 1)$. The transformation from binary to polybinary takes place in two steps. First, the original sequence $\{x_k\}$, consisting of binary ones and zeros, is converted into another binary sequence $\{y_k\}$,

as follows. The present binary digit of sequence $\{y_k\}$ is formed from the modulo-2 addition of the $(j-2)$ immediately preceding digits of sequence $\{y_k\}$ and the present digit x_k. For example, let

$$y_k = x_k \oplus y_{k-1} \oplus y_{k-2} \oplus y_{k-3} \tag{2.36}$$

Here x_k represents the input binary digit and y_k the kth encoded digit. Since the expression involves $(j-2) = 3$ bits preceding y_k, there are $j = 5$ signaling levels. Next, the binary sequence $\{y_k\}$ is transformed into a polybinary pulse train $\{z_k\}$ by adding *algebraically* the present bit of sequence $\{y_k\}$ to the $(j-2)$ preceding bits of $\{y_k\}$. Therefore, z_k modulo-2 = x_k, and the binary elements one and zero are mapped into even- and odd-valued pulses in the sequence $\{z_k\}$. Note that each digit in $\{z_k\}$ can be independently detected despite the strong correlation between bits. The primary advantage of such a signaling scheme is the redistribution of the spectral density of the original sequence $\{x_k\}$, so as to favor the low frequencies, thus improving system bandwidth efficiency.

2.10 CONCLUSION

In this chapter we have considered the first important step in any digital communication system, transforming the source information (both textual and analog) to a form that is compatible with a digital system. We treated various aspects of sampling, quantization (both uniform and nonuniform), and pulse code modulation (PCM). We considered the selection of pulse waveforms for the transmission of baseband signals through the channel. We also introduced the duobinary concept of adding a controlled amount of ISI to achieve an improvement in bandwidth efficiency at the expense of an increase in power.

REFERENCES

1. Black, H. S., *Modulation Theory,* D. Van Nostrand Company, Princeton, N.J., 1953.
2. Oppenheim, A. V., *Applications of Digital Signal Processing,* Prentice-Hall, Inc., Englewood Cliffs, N.J., 1978.
3. Stiltz, H., ed., *Aerospace Telemetry,* Vol. 1, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1961, p. 179.
4. Hecht, M., and Guida, A., "Delay Modulation," *Proc. IEEE,* vol. 57, no. 7, July 1969, pp. 1314–1316.
5. Deffebach, H. L., and Frost, W. O., "A Survey of Digital Baseband Signaling Techniques," *NASA Technical Memorandum NASATM X-64615,* June 30, 1971.
6. Lender, A., "The Duobinary Technique for High Speed Data Transmission," *IEEE Trans. Commun. Electron.,* vol. 82, May 1963, pp. 214–218.
7. Lender, A., "Correlative (Partial Response) Techniques and Applications to Digital Radio Systems," in K. Feher, *Digital Communications: Microwave Applications,* Prentice-Hall, Inc., Englewood Cliffs, N.J., 1981, Chap. 7.

8. Couch, L. W., II, *Digital and Analog Communication Systems,* Macmillan Publishing Company, New York, 1982.
9. Lender, A., "Correlative Digital Communication Techniques," *IEEE Trans. Commun. Technol.,* Dec. 1964, pp. 128–135.

PROBLEMS

2.1. You want to transmit the word "HOW" using an 8-ary system.
- **(a)** Encode the word "HOW" into a sequence of bits, using 7-bit ASCII coding, followed by an eighth bit for error detection, per character. The eighth bit is chosen so that the number of ones in the 8 bits is an even number. How many total bits are there in the message?
- **(b)** Partition the bit stream into $k = 3$ bit segments. Represent each of the 3-bit segments as an octal number (symbol). How many octal symbols are there in the message?
- **(c)** If the system were designed with 16-ary modulation, how many symbols would be used to represent the word "HOW"?
- **(d)** If the system were designed with 256-ary modulation, how many symbols would be used to represent the word "HOW"?

2.2. We want to transmit 800 characters/s, where each character is represented by its 7-bit ASCII codeword, followed by an eighth bit for error detection, per character, as in Problem 2.1. A multilevel PAM waveform with $M = 16$ levels is used.
- **(a)** What is the effective transmitted bit rate?
- **(b)** What is the symbol rate?

2.3. We wish to transmit a 100-character alphanumeric message in 2 s, using 7-bit ASCII coding, followed by an eighth bit for error detection, per character, as in Problem 2.1. A multilevel PAM waveform with $M = 32$ levels is used.
- **(a)** Calculate the effective transmitted bit rate and the symbol rate.
- **(b)** Repeat part (a) for 16-level PAM, eight-level PAM, four-level PAM, and PCM (binary) waveforms.

2.4. Given an analog waveform that has been sampled at its Nyquist rate, f_s, using natural sampling, prove that a waveform (proportional to the original waveform) can be recovered from the samples, using the recovery techniques shown in Figure P2.1. The parameter mf_s is the frequency of the local oscillator, where m is an integer.

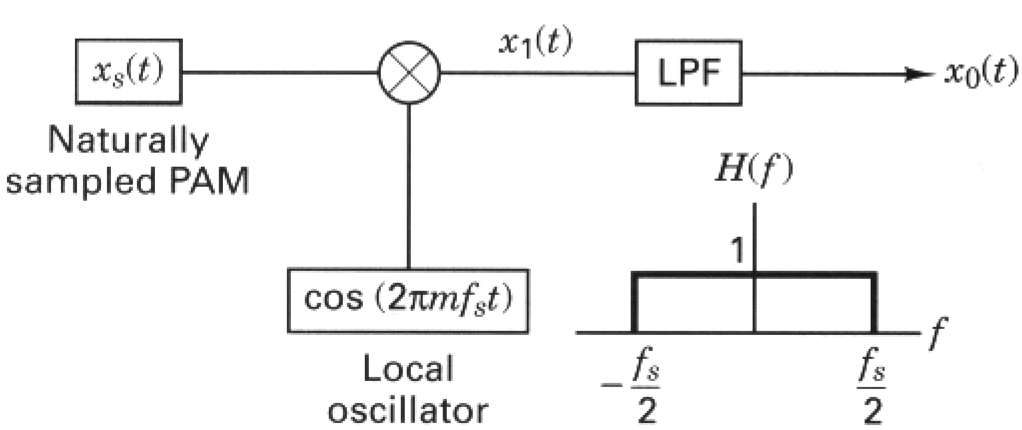


Figure P2.1

2.5. An analog signal is sampled at its Nyquist rate $1/T_s$, and quantized using L quantization levels. The derived digital signal is then transmitted on some channel.

(a) Show that the time duration, T, of one bit of the transmitted binary encoded signal must satisfy $T \leq T_s/(\log_2 L)$.

(b) When is the equality sign valid?

2.6. Determine the number of quantization levels that are implied if the number of bits per sample in a given PCM code is (a) 5; (b) 8; (c) x.

2.7. Determine the minimum sampling rate necessary to sample and perfectly reconstruct the signal $x(t) = \sin(6280t)/(6280t)$.

2.8. Consider an audio signal with spectral components limited to the frequency band 300 to 3300 Hz. Assume that a sampling rate of 8000 samples/s will be used to generate a PCM signal. Assume that the ratio of peak signal power to average quantization noise power at the output needs to be 30 dB.

(a) What is the minimum number of uniform quantization levels needed, and what is the minimum number of bits per sample needed?

(b) Calculate the system bandwidth (as specified by the main spectral lobe of the signal) required for the detection of such a PCM signal.

2.9. A waveform, $x(t) = 10\cos(1000t + \pi/3) + 20\cos(2000t + \pi/6)$ is to be uniformly sampled for digital transmission.

(a) What is the maximum allowable time interval between sample values that will ensure perfect signal reproduction?

(b) If we want to reproduce 1 hour of this waveform, how many sample values need to be stored?

2.10. **(a)** A waveform that is bandlimited to 50 kHz is sampled every 10 μs. Show graphically that these samples uniquely characterize the waveform. (Use a sinusoidal example for simplicity. Avoid sampling at points where the waveform equals zero.)

(b) If samples are taken 30 μs apart instead of 10 μs, show graphically that waveforms other than the original can be characterized by the samples.

2.11. Use the method of convolution to illustrate the effect of undersampling the waveform $x(t) = \cos 2\pi f_0 t$ for a sampling rate of $f_s = \frac{3}{2} f_0$.

2.12. Aliasing will not occur if the sampling rate is greater than twice the signal bandwidth. However, perfectly bandlimited signals do not occur in nature. Hence, there is always some aliasing present.

(a) Suppose that a filtered signal has a spectrum described by a Butterworth filter with order $n = 6$, and upper cutoff frequency $f_u = 1000$ Hz. What sampling rate is required so that aliasing is reduced to the −50 dB point in the power spectrum?

(b) Repeat for a Butterworth filter with order $n = 12$.

213. **(a)** Sketch the complete $\mu = 10$ compression characteristic that will handle input voltages in the range −5 to +5 V.

(b) Plot the corresponding expansion characteristic.

(c) Draw a 16-level nonuniform quantizer characteristic that corresponds to the $\mu = 10$ compression characteristic.

2.14. The information in an analog waveform, whose maximum frequency $f_m = 4000$ Hz, is to be transmitted using a 16-level PAM system. The quantization distortion must not exceed ±1% of the peak-to-peak analog signal.

(a) What is the minimum number of bits per sample or bits per PCM word that should be used in this PAM transmission system?

(b) What is the minimum required sampling rate, and what is the resulting bit rate?

(c) What is the 16-ary PAM symbol transmission rate?

2.15. A signal in the frequency range 300 to 3300 Hz is limited to a peak-to-peak swing of 10 V. It is sampled at 8000 samples/s and the samples are quantized to 64 evenly spaced levels. Calculate and compare the bandwidths and ratio of peak signal power to rms quantization noise if the quantized samples are transmitted either as binary pulses or as four-level pulses. Assume that the system bandwidth is defined by the main spectral lobe of the signal.

2.16. In the compact disc (CD) digital audio system, an analog signal is digitized so that the ratio of the peak-signal power to the peak-quantization noise power is at least 96 dB. The sampling rate is 44.1 kilosamples/s.

(a) How many quantization levels of the analog signal are needed for $(S/N_q)_{\text{peak}} =$ 96 dB?

(b) How many bits per sample are needed for the number of levels found in part (a)?

(c) What is the data rate in bits/s?

2.17. Calculate the difference in required signal power between two PCM waveforms, NRZ and RZ, assuming that each signaling scheme has the same requirements for data-rate and bit-error probability. Also assume equally likely signaling, and that the difference between the high-voltage and low-voltage levels is the same for both the NRZ and RZ schemes. If there is a power advantage in using one of the signaling schemes, what, if any, is the disadvantage in using it?

2.18. In the year 1962, AT&T first offered digital telephone transmission referred to as T1 service. With this service, each T1 frame is partitioned into 24 channels or time slots. Each time slot contains 8 bits (one speech sample), and there is one additional bit per frame for alignment. The frame is sampled at the Nyquist rate of 8000 samples/s, and the bandwidth used for transmitting the composite signal is 386 kHz. Find the bandwidth efficiency (bits/s/Hz) for this signaling scheme.

2.19. **(a)** Consider that you desire a digital transmission system, such that the quantization distortion of any audio source does not exceed $\pm 2\%$ of the peak-to-peak analog signal voltage. If the audio signal bandwidth and the allowable transmission bandwidth are each 4000 Hz, and sampling takes place at the Nyquist rate, what value of bandwidth efficiency (bits/s/Hz) is required?

(b) Repeat part (a) except that the audio signal bandwidth is 20 kHz (high fidelity), yet the available transmission bandwidth is still 4000 Hz.

QUESTIONS

2.1 What are the similarities and differences between the terms "formatting" and "source coding"? (See Chapter 2, introduction.)

2.2 In the process of *formatting* information, why is it often desirable to perform *oversampling?* (See Section 2.4.3.)

2.3 In using pulse code modulation (PCM) for digitizing analog information, explain how the parameters *fidelity, bandwidth,* and *time delay* can be traded off. (See Section 2.6.)

2.4 Why is it often preferred to use units of normalized bandwidth, *WT* (or time-bandwidth product), compared with bandwidth alone? (See Section 2.8.3.)

EXERCISES

Using the Companion CD, run the exercises associated with Chapter 2.